3rd MATHMOD
VIENNA

3rd MATHMOD
VIENNA

3rd MATHMOD
VIENNA

PROCEEDINGS

IMACS Symposium on
MATHEMATICAL MODELLING
February 2-4, 2000
Vienna University of Technology, Austria

I. Troch
F. Breitenecker
Editors

ARGESIM Report 16

ASIM Mitteilung 72B

ISBN print: 978-3-901608-15-5

ISBN e-book: 978-3-901608-16-2

Reihe „ARGESIM Reports“
Reihenherausgeber: F. Breitenecker

ARGESIM Report No. 16:
Proceedings
IMACS Symposium on MATHEMATICAL MODELLING
February 2 – 4, 2000

Herausgeber: I. Troch, F. Breitenecker

ARGE Simulation News (ARGESIM)
Dept. Simulation (E1143)
Vienna University of Technology
Wiedner Hauptstraße 8-10
A-1040 Vienna, Austria
Tel.: +43-1-58801-11452
Fax: +43-1-58801-11499
WWW: http://www.argesim.org

ARGESIM Report 15

ASIM Mitteilung 72B

ISBN print: 978-3-901608-15-5

ISBN e-book: 978-3-901608-16-2

Preface

The possibility to solve certain problems and the quality of a solution obtained for a specific task depend essentially on appropriate modelling of the task in question. In many fields of application this leads to the necessity to model the dynamic and static properties of a system to be constructed, to be improved or to be influenced, but also all relevant background information connected with this task. Normally, it will not be possible to carry out all this modelling as requested – partly because not all information is available and partly because the resulting model of the respective task would be of size which can not be handled by men or computers. Hence, it is always the decision of the modeller what is to be included in the model and whether a model of the complete task is to be created or e.g 'only' a model of the dynamic behaviour of the system to be investigated or to be influenced.

Moreover, in some cases, the system under investigation and its behaviour are understood rather well. In such cases an appropriate model of the system's behaviour will assist in finding a good solution of the problem to be solved. In other situations a model of the system's behaviour is primarily intended to help for a better understanding of what is going on in the system. Examples for the first case are many types of design problems connected typically with engineering sytems, such as controller design, design of a production line etc. whereas the request for an improved understanding is often found in connections with non-engineering systems such as biological or medical systems, economic or environmental systems and their control etc.

There is a rather wide consensus that mathematical modelling i. e. abstraction and formalization, is of intrinsic importance. Moreover, most engineers and scientists know quite well that appropriate modelling is far from being easy and that the quality of a design depends strongly on the quality of the model. One of the most important challenges connected with proper modelling is the request to model indeed the given task i. e. all relevant information, restrictions, demands, goals etc. In control engineering not only a model of the plant and constraints on relevant physical variables must be put in a mathematical form but also other requests such as that the resulting mathematical control law must allow for implementation by means of a certain type of equipment etc.

By now, considerations such as these are accepted in general and especially by those involved in the solution of problems by using computers what means by using – directly or indirectly – mathematical methods, no matter whether these persons work at a scientific institution or in an industrial environment.

However, the area of application determines to a certain extent the knowledge of basic modelling principles, preferences for modelling approaches, for methods for model simplification or for parameter estimation etc. Moreover, many things are discovered repeatedly. Therefore, a conference having mathematical modelling as its focus will allow for a fruitful and stimulating exchange of ideas. Consequently, the third IMACS Symposium on Mathematical Modelling (3rd MATHMOD) is devoted to the mathematical (or formal) modelling of all type of systems no matter whether the system is

* dynamic or static
* deterministic or stochastic
* continuous or discrete
* lumped parameter or distributed parameter
* linear or nonlinear
* or of any other nature.

Thus, a wide variety of formal models is to be discussed and the term "mathematical model" includes classical models such as differential or difference equations, Markov processes, ARMA models as well as more special or more recent approaches such as Bond graphs, Petri nets, fuzzy models or neural nets.

The written version of the contributions to 3rd MATHMOD Vienna are collected in these Proceedings starting with the manuscripts of the invited lectures. The first survey to be presented is concerned with a problem of great actuality, i.e. with the question of how to development of intuition. This talk was motivated by the observation that skelled engineers often use their intuition to solve dynamic problems e.g. by modifying components of a system. On the other hand, we live in an era of big simulation programs based on a variety of submodels and understanding of the assumptions inherent throughout the system is impossible and consequently, simulation often becomes difficult or the suggested solution is not of the desired quality. Hence, development of intuition is more important than ever and the author discusses how Bond graphs can serve as tools to develop a better physical understanding.

Interconnected dynamical systems are also one of the main issues of the second plenary lecture which has as aim to introduce some of the main features of the behavioral approach as a mathematical language for discussing mathematical models. Here, a model is viewed as an exclusion law, and the behavior of the model is that subset of outcomes of the systems that the model declares possible. Some results about elimination are formulated in the context of systems described by linear ordinary and partial differential differential equations and important concepts such as controllability and observability are discussed in this setting.

The third survey is concerned with the inverse simulation approach i.e. with models which allow determination of the time history of 'inputs' needed to achieve a specific time history for a selected set of 'outputs'. Methods and algorithms are reviewed, which the author believes to have wide applicability across a number of fields and which have received attention in variousfields of application. Moreover, the potential of this approach for the important external validation of a simulation model is emphasized.

Computer algebra systems become more and more important for solving real-life problems by using modelling and simulation. Nowadays, their power allows to handle very complex (nonlinear) systems analytically i.e. symbollically so that solutions may be gained without time-consuming calculations in other domains (e.g. time domain). This holds also for the determination of optimal parameters in distributed parameter models, what is also shown in this forth survey.

The last surveys highlights the influence of the modelling goal on the resulting model's complexity. This lecture emphasizes the particularties of models for fault detection and isolation (FDI) in contrast to models used for control. Of special interest is the question of complexity of the model, which, of course, depends basically on the given situation. However, it is shown that FDI-models – contrary to widespread opinions – may be even less complex and precise because they can restrict to only those parts of the system in which the faults occur and to those phenomena that carry information of the faults of interest. This issue is discussed in terms of different model-based FDI approaches – analytical, knowledge-based and data-based.
Then follow groups of papers which were contributed upon invitation of a session organizer and, contributed papers selected for presentation after a reviewing process which was based on extended abstracts. All these contributions were colleted and arranged in sessions according to their main thematic point. Such a grouping is by no means easy because many contributions address several different aspects in a balanced manner. Therefore, the arrangement chosen for this volume follows rather closely the one of the conference where also time limitations had to be observed.

The editors wish to express their sincere thanks to all who have assisted them by making the idea of this symposium known within the scientific community or by acting as sponsor or cosponsor, who have assisted them in the reviewing process and have done a good job by putting together special sessions devoted to one main theme. Last but not least the editors thank Creditanstalt-Bankverein for their generous support for the production of these Proceedings.

Vienna, January 2000 I.Troch, F. Breitenecker

Contents

Invited Lectures

Engineering Applications

Human-Machine Interfaces in Robotics

Special Session organized by S. P. Tzafestas (Athens, GR)

Virtual Reality and Simulation in Robotics

Special Session organized by S. P. Tzafestas (Athens, GR)

Contributed papers:

Descriptor Systems

Methods and Theoretical Aspects

Discrete Systems

Software and Softcomputing

Virtual Reality in Modeling and Simulation

Special Session organized by D. P. F. Möller (Hamburg, D)

Contributed papers:

Biology and Medicine

Some Challenging Topics in PK-PD Modelling

Special Session organized by R. Karba (Ljubljana, SI)

Education

Biotechnical and Chemical Engineering

Bondgraphs

Bondgraph modelling

Special Session organized by J. Thoma (Zug, CH)

Control Systems

Modelling for Control and Supervision

Special Session organized by B. Zupancic (Ljubljana, SI)

Mechanics and Mechatronics, incl. Robotics

Retaining Analog Intuition in a Digital World with Bond Graphs

Dean Karnopp
Dept. of Mechanical and Aeronautical Engineering
The University of California
Davis, CA 95616, USA
dckarnopp@ucdavis.edu

Abstract. Although digital computers have supplanted analog computers for the simulation of physical systems, and digital signal processors have largely replaced analog devices in control systems, most engineers still think about mechatronic systems in analog terms. Skilled circuit designers and many others can use their intuition to solve dynamic problems by modifying components or inserting new elements into a system. This intuition is based not only on experience with physical systems, but also on simulation of continuous signal mathematical models.

Digital computers allow the construction of extremely complex mathematical models by the combination of a variety of submodels, but it often happens that an intuitive understanding of the assumptions inherent throughout the system is nearly impossible, so that if the simulation is difficult or the predicted performance is poor, the remedy is far from obvious. Bond graph based models allow a system analyst to think in physical terms about system components and to appreciate the mathematical and computational aspects of changes in subsystem elements or parameters. Bond graph processors and programs containing libraries of submodels make the process of translation from a physically based model, to a mathematical model, and then to a computational simulation scheme nearly automatic. In this way, the system designer's intuition can be brought to bear in solving dynamic problems in mechatronic systems.

Introduction.

To those who have come of age in an era of ever increasing digital computer power, at an ever decreasing cost, it may seem surprising that complex dynamic systems were successfully designed in the past using only limited analytical methods and primitive computational aids. It is a testament to the power of the human being to discern patterns, and to develop intuition about complex system behavior, that many successes were achieved albeit not without a number of sometimes spectacular failures. It certainly cannot be denied that modern digital hardware now allows very sophisticated control of dynamic systems, but the attempt to mathematically model and simulate ever more complex systems in the digital domain is not as straight forward and rapid as desired. In some cases, it is clear that human intuition gained by experience with physical systems and with simple models in the form of circuit diagrams, mechanical schematic diagrams, or even analog computers cannot be easily utilized in modifying a simulation model once it has been rendered into digital form. An engineer who might successfully tinker with a physical prototype system to improve its behavior is often baffled by a computer model, which predicts unsatisfactory performance. The purpose of this paper is to point out aspects of bond graph modeling, which allow an engineer to think about a dynamic system in physical and analog terms, even if the system will be simulated and its controller will be realized in purely digital form. In this way, human intuition about physical systems may be applied to the construction of a reasonable system model and physical modifications to components to improve performance can be readily transferred to the computational model.

Megamodel Projects.

With the increase in computer capability there naturally arises the desire to construct larger, more accurate models of complex multi-domain systems. An example, is the TOOLSYS project involving automobile manufacturers and suppliers, [1]. The idea is to create simulation models by combining submodels independently developed using a variety of tools and proprietary languages each tailored to specific types of components. Figure 1 taken from Ref. [1] shows an example system involving COMPAMM, a multi-body mechanics program [2]. AMESim, a language for hydraulic systems, [3] and ADVance-MS, for electrical systems, [4] all coordinated by VHDL-AMS, a new standard also called IEEE

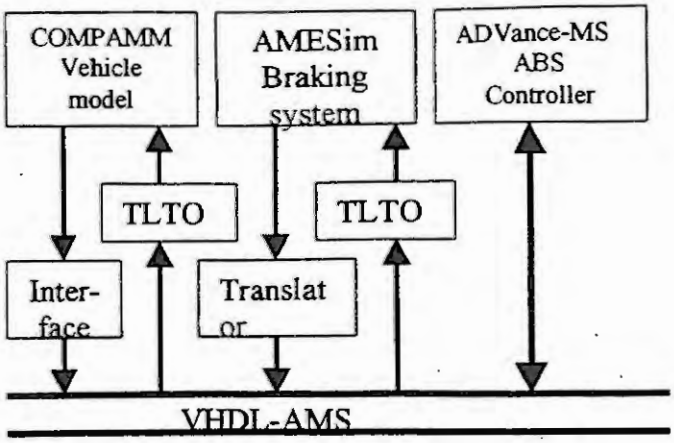


Fig. 1 Architecture of the integrated TOOLSYS environment

1076.1, which can represent discrete and continuous time systems. Since submodels may be developed by separate, competing organizations, the submodels may be provided only in compiled form to disguise parameter values and other details of proprietary subsystems. The idea of complete system models capable of validating the performance of complex multi-domain digitally controlled systems is attractive, but the problems of coordinating disparate subsystem models in a variety of digital languages is formidable and an intuitive understanding of the behavior of the entire system will certainly be nearly impossible. Thus, if problems arise, it will be hard for anyone to bring dynamic system experience to bear in solving the problems.

In the following, we discuss the more modest goal of using bond graphs as a basis for creating multi-domain models, with a clear connection to physical and analog concepts. The mathematical models may ultimately be simulated digitally and the actual system will almost certainly be controlled by digital hardware, but the connection to the physical world will be maintained.

Bond Graphs, Circuits, and Schematics.

A universal step in the design of any physical system is a graphical representation of components and their connections. At the simplest level, electrical circuits and mechanical or hydraulic schematic diagrams essentially represent a model of a proposed system and with experience engineers can often say quite a bit about the dynamics of the system just by looking at these abstract diagrams.

Bond graphs [5] are in one sense generalizations of this type of diagram. Indeed, the standard procedure for turning an electric circuit into a bond graph starts by reproducing the circuit topology exactly with nodes represented by 0-junctions and the circuit elements strung between the nodes on 1-junctions. Hydraulic circuits can be handled similarly, while schematic diagrams for mechanical systems, which are less obviously circuit-like, are also readily converted to bond graph form. For multi-domain systems, the diagrams are often less satisfactory than a corresponding bond graph. For example, Fig. 2 shows a somewhat old fashioned schematic diagram and an equivalent circuit for a dc motor [6]. The circuit is nondynamic and the mechanical port is represented only by the back voltage E_a. The bond graph on the other hand can be augmented to include dynamic effects related to inductance, mechanical inertia and friction and can indicate that the flux in the field is related to the two circuits I_f and I_a.

The bond graph, in its non causal form, represents a physically based model. It is capable of generating multiple sets of equations depending upon the causality assumed at the electrical and mechanical ports, and upon the modelers assumptions about the presence or absence of certain elements in the graph [7].

Bond graphs also provide circuit-like depictions of far more complex systems for which no simple schematic diagram is known. Figure 3, for example, shows a model of a long flexible bar moving through large angles and simultaneously vibrating. The bond graph represents the structure of the model equations, which include three rigid body nodes and two bending nodes [8]. Using this type of bond graph, an approach to modeling complex mechanical systems is rendered understandable to anyone with some background in vibrations.

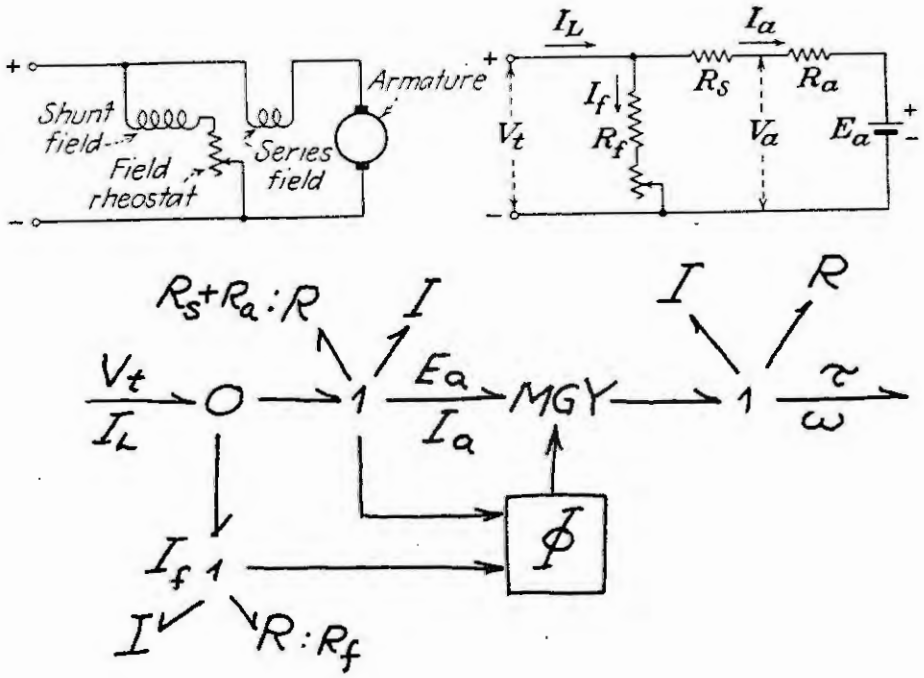


Fig. 2 Schematic diagram, equivalent circuit and bond graph for compound wound dc motor

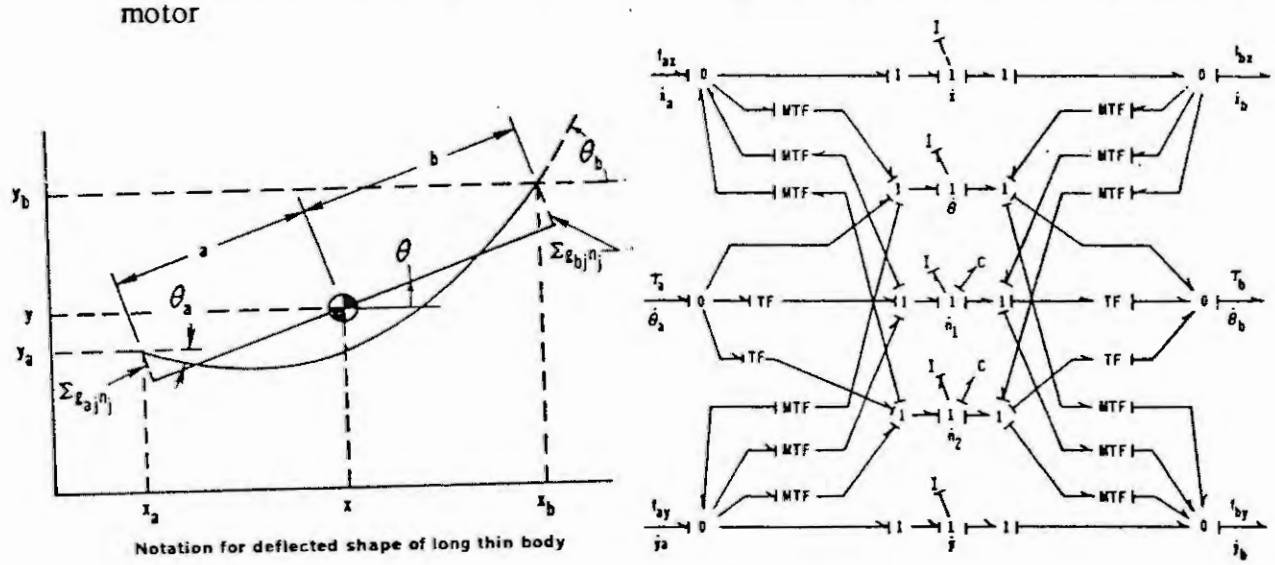


Fig. 3 Sketch and bond graph for a translating rotating and vibrating thin body

Bond Graphs and Block Diagrams.

Once causality has been applied to a bond graph it becomes equivalent to a condensed version of a detailed block diagram, i.e., all input-output signals have been determined throughout the system. Many control engineers understand block diagrams better than sets of equations so this feature of bond graphs can help them to understand a model. As the example in Fig. 4 shows, the block diagram derived from a bond graph makes plain the back and forth reaction effects so typical of physical systems as contrasted to the one-way information transfer often encountered in control systems.

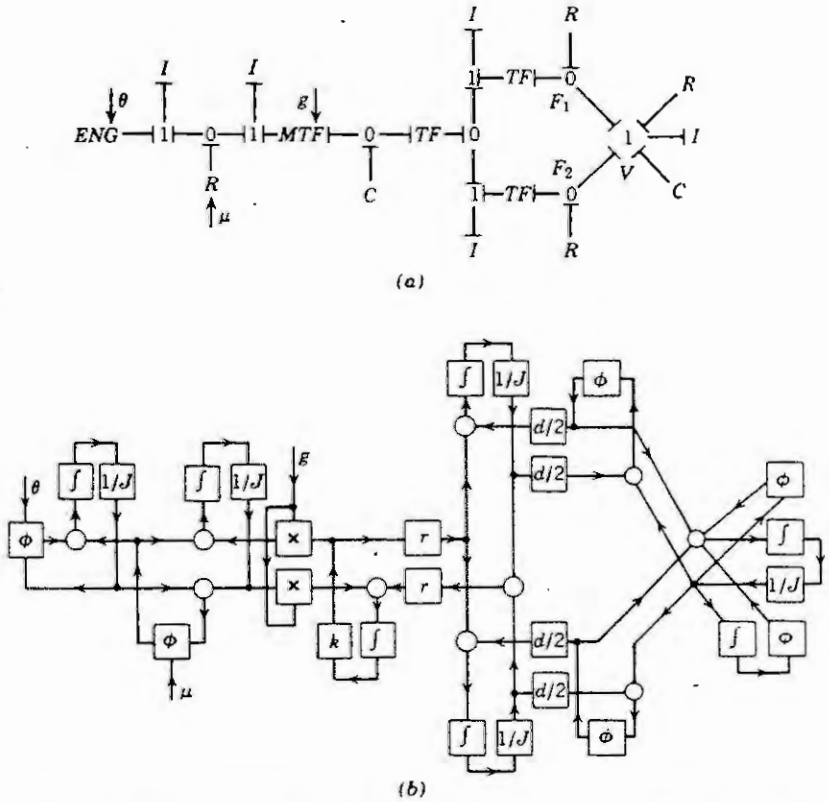


Fig. 4 Drive train model. (a) bond graph; (b) block diagram

Figure 5 shows a simple analog computer designed by the author more than 35 years ago for use in hardware-in-the-loop testing of digital steam turbine speed controllers. In this case, the block diagram model of the plant is obvious from the faceplate and the setting of parameters was equally clear. In those simpler days, intuition about dynamic systems was often based on block diagrams and analog computers. To some extent bond graphs extend this experience to far more complex system models.

Bond Graph Causality and Model Consistency.

Bond graph models of components are fundamentally non causal, meaning that any external port, either an effort variable or a flow variable may serve as an input and the model will respond with the complementary power variable. In principle, this means that the model may produce a variety of state equations depending upon how it interacts with the remainder of the system [7]. Most attempts to produce libraries of predetermined component models do not have this flexibility. Figure 6, for example, represents an "encrypted template" for a dc motor from the Saber Automotive Template library [9]. This model assumes that a voltage is applied to the terminals p and m and the output variable is the speed w. The bond graph shows the model with the assumed causality and the parameters. If the motor is driving a load, there also should be a load torque as an input, and the current should be an extra output. With an input-output type of subsystem representation, it is easy to be seduced by the idea that there is a simple dynamic relation between voltage and speed, but this really is incomplete, neglecting possible back effects associated with current draw and load torque.

The intelligent use of causal models such as those in Saber will often produce useful system models. However, there is the danger that back effects physically present may inadvertently neglected or that models will be assembled which cannot function without some kind of trick. Two motors as represented in Fig. 6 cannot be geared together without including some compliance between them even if the mechanical coupling is extremely stiff. This could lead to the construction of a numerically stiff model for a system, which could be much more intelligently modeled if the problem were better understood. One of the most important uses of bond graph causality is to investigate and solve such problems before a simulation is attempted.

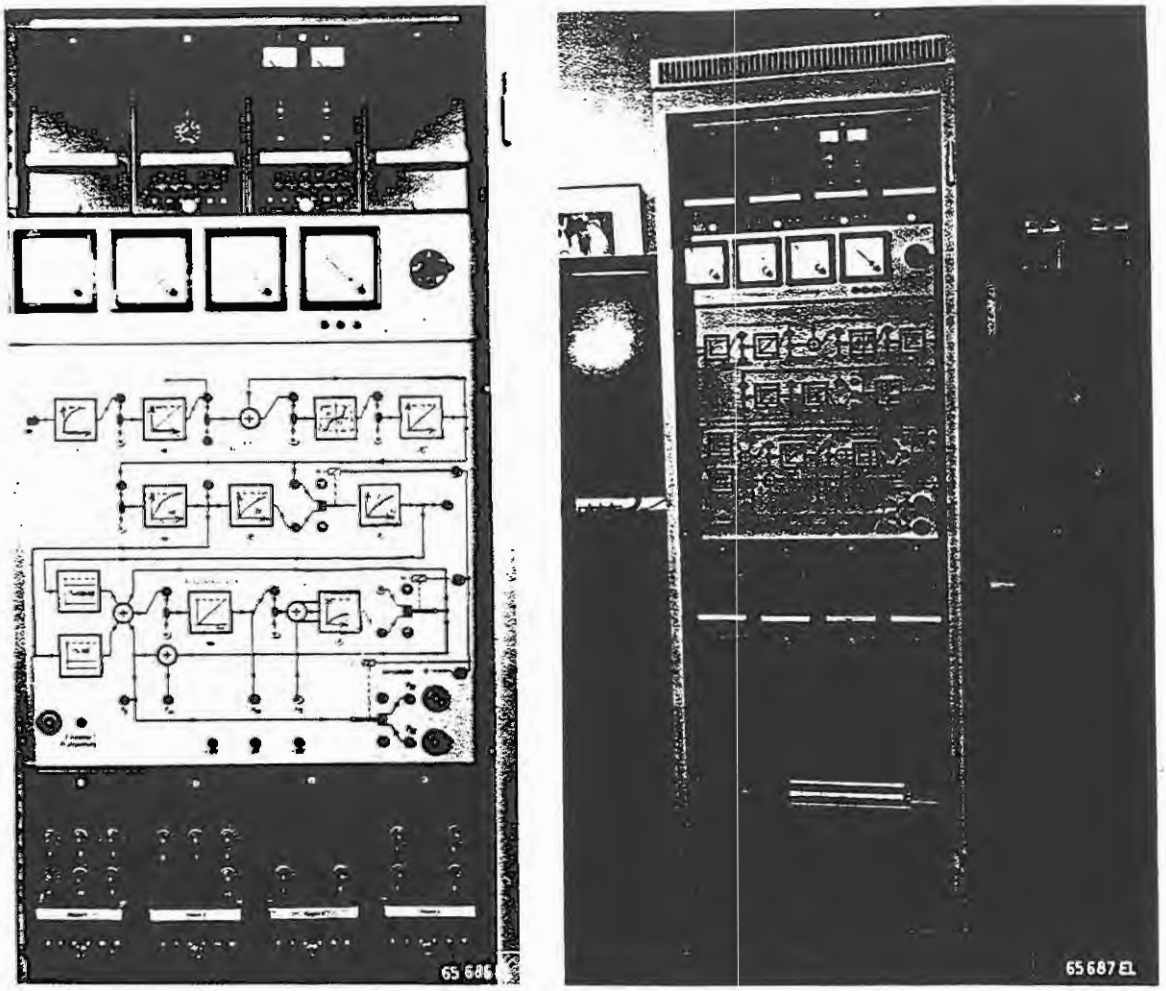
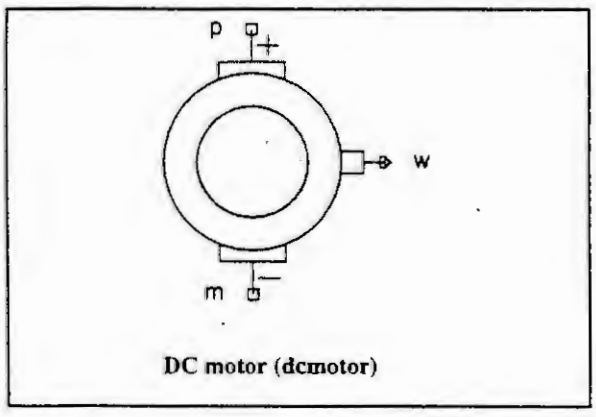


Fig. 5 Block diagram based analog simulator of steam turbine



dcmotor (DC Motor)

encrypted template dcmotor p m w = kt, ke, l, r, j, d, units

The **dcmotor** template models a DC motor with electrical input connections (**p** and **m**), and an output (**w**) that is a **var** with no units. This output value represents the angular velocity of the shaft and can be used as input to other Control System templates. The motor polarity is such that a positive voltage applied from **p** to **m** will produce a positive angular velocity.

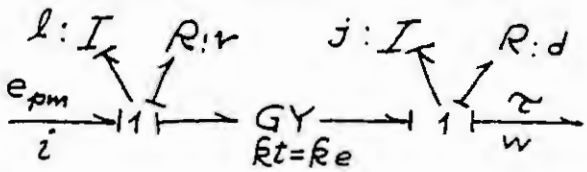


Fig. 6 DC Motor template from Saber and equivalent bond graph

Bond Graphs and Component Icons.

It has long been appreciated that the attempt to make a bond graph model of a component is particularly useful way to study the physics of the system and to appreciate how the component will interact with other parts of a system. However, there has been a reluctance to use bond graphs for complex systems in industry. The idea is that many engineers cannot be expected to deal with a complex system involving several energy domains if they have not had a thorough education in bond graph modeling. (For those specializing in physical system modeling, the existence of bond graph processors such as CAMP-G [10] allows an easy transition from an overall bond graph model to input code for any of a number of simulation programs.)

A possible solution to this problem is to create a library of bond graph models rather than input-output models and to represent them as icons similar to those used in engineering practice. This procedure has been carried out for hydraulic and mechanical systems in AMESim [3] [11]. Figure 7 shows a typical element which is represented by an icon on a computer screen, but which is based on a bond graph submodel. If complex systems are created using the icons, the user can go back to individual parts of the system to see the bond graph of the elements and to consider physical changes in the system to improve performance. The intuitive ideas experienced hydraulic engineers might have about adding components or modifying existing components can thus be readily implemented in the model. Furthermore, a systems analyst can appreciate the mathematical implications of these changes by considering the bond graphs of the affected parts.

In fact, AMESim is a part of the TOOLSIS project mentioned previously and there is little doubt that a bond graph based icon system could also be developed for electrical and electromechanical components. For some types of systems at least, this procedure might have a better chance of success than the attempt to combine a series of models developed using very different modeling approaches and languages. Beyond an extention to more types of elements and to new energy domains, there is still work to be done in allowing flexibility in causality. Note that in Fig. 7, the bond graph is given with fixed port causality. This will limit which icons can be successfully joined. One could imagine the imposition of causality only after the system was completed although this would require the system modeler to handle algebraic loops and derivative causality in some manner if they happened to arise. (Such problems must be addressed by any modeling technique. They are not unique to bond graph methods.)

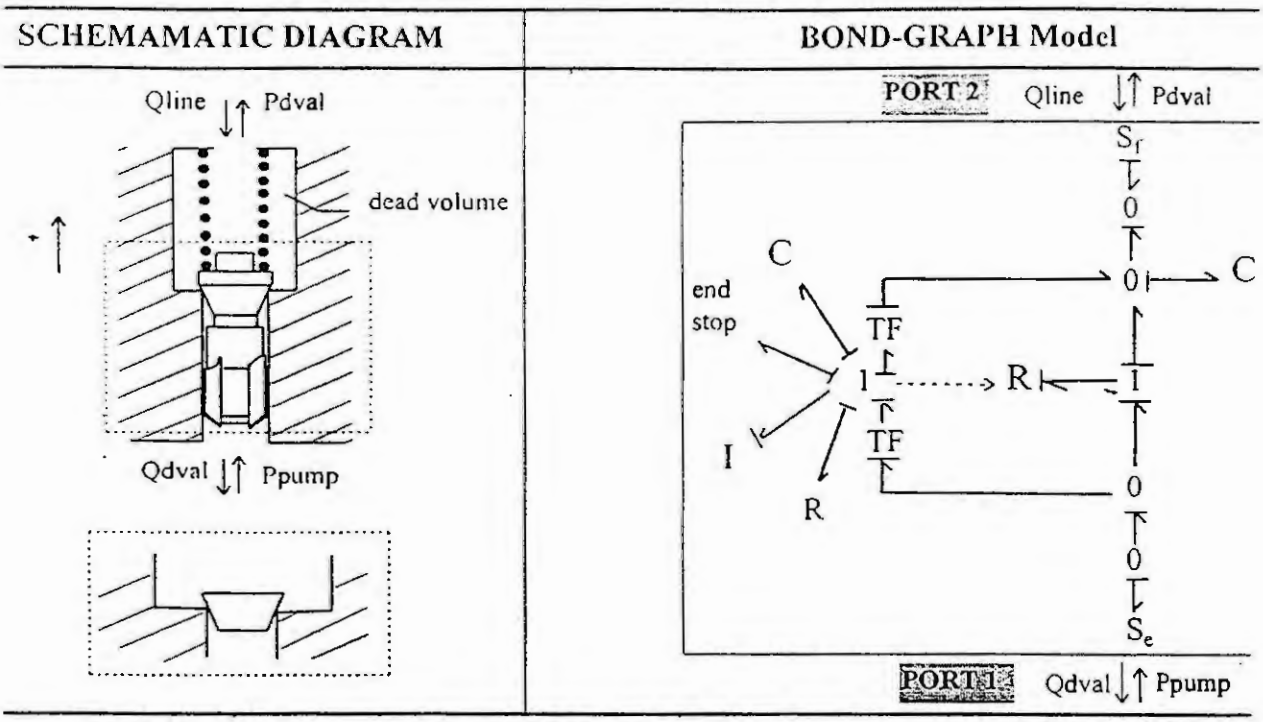


Fig. 7 Model of fuel injection delivery valve in AIMSim

Conclusions.

It is not an easy task to automate the modeling and simulation of multi-domain dynamic systems without losing the intuitive feeling that experienced engineers have for physical systems. Despite the criticism that bond graphs seem too abstract to beginners, they have the potential to keep the physical basis of the model in the forefront, and by the use of bond graph-based icons, complex systems can be described in an energetically correct manner even by non-specialists. By focusing on physics rather than on digital models, engineers can make good use of their experience and intuition.

References

[1] Andersson, M. and Tirgari, P., Simulation of Vehicle Braking Stability — A Model Integration Case Study, Proc. European Simulation Symposium, Erlangen, Germany, 1999, pp. 137-141.

[2] Jimenez, A., Garcia-Alonso, G., "COMPAMM: A Simple and Efficient Code for Kinematic and Dynamic Numerical Simulation of 3D Systems with Realistic Graphics" in Multibody Systems Handbook, W. Schielen, ed., Springer-Verlag, 1990, pp. 285-304.

[3] AMESim, Imagine Investissements, 5 rue Brison, 42300 Roane, France.

[4] Mentor Graphics, http://www.mentorg.com/ams/

[5] Karnopp, D.C., Margolis, D.L., and Rosenberg, R.C., System Dynamics: Modeling and Simulation of Mechatronic Systems, John Wiley & Sons, New York, in press.

[6] Fitzgerald, A.E., Basic Electrical Engineering, McGraw-Hill, NY, 1945.

[7] Karnopp, D., Structure in Dynamic System Models—Why a Bond Graph is more Informative than its Equations, Proc. 12th IMACS World Congress, Paris, 1988, pp. 1-4.

[8] Margolis, D.L. and Karnopp, D.C., Bond Graphs for Flexible Multibody Systems, Trans. ASME, J. of Dyn. Sys. Meas. and Cont. N.101, n1, 1979, pp. 50-57.

[9] Saber, Analog Inc., 9205 SW Gemini Drive, Beaverton, OR 97005-7156, USA.

[10] CAMP-G, Computer Aided Modeling Program, CADSIM Engineering, P.O.Box 4083, Davis, CA 95617, USA, grandajj@ecs.csus.edu.

[11]Lebrun, M. and Richards, C., How to Create Good Models without Writing a Single Line of Code, 14 p. Proceedings, Fifth Scandinavian International Conference on Fluid Power, Linkoping, Sweden, 1997.

Modelling Dynamical Systems using Manifest and Latent Variables

Jan C. Willems
Mathematics Institute, University of Groningen
P.O. Box 800, 9700 AV Groningen, The Netherlands
e-mail: J.C.Willems@math.rug.nl.

Abstract. The behavioral approach provides a mathematical language for the modeling of systems, particularly dynamical systems. An introduction to behaviors is given, with emphasis to interconnected systems. This is viewed as consisting of modules, combined with an interconnection architecture. The latter is formalized as a graph with leaves. The elimination theorem is discussed. This allows to obtain behavioral equations involving only manifest variables, starting from models that contain also latent variables. Subsequently, the notions of controllability and observability are cast in this setting.

1 Introduction

The purpose of this presentation is to outline the basics of a mathematical language for the modeling, analysis, and the synthesis of dynamical systems. The framework that we will present considers the *behavior* of a system as the main object of study. This paradigm differs in an essential way from the input/output paradigm which has dominated the development of the field of systems and control in the 20-th century. This paradigm-shift calls for a reconsideration of many of the basic concepts, of the model classes, of the problem formulations, and of the algorithms in the field.

It is impossible to do justice to all these aspects in the span of a one hour presentation. We will therefore concentrate of a few main themes:

- The basic motivation, in the context of modeling, of the conceptual framework that is used.
- The role of latent variables as they emerge from modeling interconnected systems.
- A discussion of system representations, mainly in the context of systems described by differential equations.
- The notions of controllability and observability in this new setting.
- The formulation of control questions and issues of implementation and design.

This article sketches a mathematical framework that allows to discuss systems in interaction with their environment. However, it is not the purpose to develop mathematical ideas for their own sake. To the contrary, we will downplay mathematical issues throughout. The main aim is to convince the reader that the behavioral framework is a cogent systems theoretic setting that properly deals with physical systems and that approaches modeling as an essential motivation for choosing appropriate concepts.

The behavioral approach is discussed, including the mathematical technicalities, in the recent textbook [1]. A very early reference that contain some of the (immature) ideas is [2]. The three part paper [3] provides the first detailed presentation of the behavior framework. It has been further elaborated in [4] and in [5]. This latter reference contains a comprehensive overview. In [6], control is discussed from this perspective. Finally, we mention the article [7] where many of these results are generalized to partial differential equations. Informal expositions of the behavioral approach can be found in [8].

2 The behavior

The framework that we use for discussing mathematical models views a model as follows. Assume that we have a phenomenon (that is, a set of outcomes) that we try to model. Nature (that is, the reality that governs this phenomenon) can produce certain outcomes. The totality of these possible outcomes (*before* we have modeled the phenomenon) forms a set $\mathbb{U}$, called the *universum*. A *mathematical model* restricts the outcomes that a model declares possible to a subset $\mathfrak{B}$ of $\mathbb{U}$; $\mathfrak{B}$ is called the *behavior* of the model. We often refer to $(\mathbb{U}, \mathfrak{B})$ as a mathematical model.

In the study of dynamical systems we are, more specifically, interested in situations where the outcomes of the phenomena are signals, i.e., maps with independent variables (time, or space, or time and space) and dependent variables (the space where the signals take on their values). In this case the universum is therefore the space of all maps from the set of independent variables to the set of dependent variables. It is hence convenient to distinguish these sets explicitly in the notation: $\mathbb{T}$ (suggesting 'time') for the set of independent, and $\mathbb{W}$ for the set of dependent variables. Whence we define a *dynamical system* as a triple $\Sigma = (\mathbb{W}, \mathbb{T}, \mathfrak{B})$ with $\mathfrak{B}$, the behavior, a subset of $\mathbb{W}^{\mathbb{T}}$, ($\mathbb{W}^{\mathbb{T}}$ is the standard mathematical notation for the set of all maps from $\mathbb{T}$ to $\mathbb{W}$).

We give a couple of examples. In the first and third $\mathbb{T}$ is time only, while in the second example, Maxwell's equations, $\mathbb{T}$ involves time and space.

1. *Newton's second law* imposes a restriction that relates the position $\vec{q}$ of a point mass and the force $\vec{F}$ acting on it. This relation is $\vec{F} = m\frac{d^2}{dt^2}\vec{q}$, with m the mass. This is a dynamical system with $\mathbb{T} = \mathbb{R}, \mathbb{W} = \mathbb{R}^3 \times \mathbb{R}^3$ (typical elements of $\mathfrak{B}$ are $w = (\vec{q}, \vec{F}) : \mathbb{R} \to \mathbb{R}^3 \times \mathbb{R}^3$) and behavior $\mathfrak{B}$ consisting of all maps $t \in \mathbb{R} \mapsto (\vec{q}, \vec{F})(t) \in \mathbb{R}^3 \times \mathbb{R}^3$ that satisfy $\vec{F} = m\frac{d^2}{dt^2}\vec{q}$. We will not specify the precise sense of what it means that a function satisfies a differential equation.

2. *Maxwell's equations* provide a typical example of a distributed dynamical system with many independent variables. They describe the possible realizations of the fields $\vec{E} : \mathbb{R} \times \mathbb{R}^3 \to \mathbb{R}^3, \vec{B} : \mathbb{R} \times \mathbb{R}^3 \to \mathbb{R}^3, \vec{j} : \mathbb{R} \times \mathbb{R}^3 \to \mathbb{R}^3$, and $\rho : \mathbb{R} \times \mathbb{R}^3 \to \mathbb{R}$. Maxwell's equations are

$$\begin{aligned} \nabla \cdot \vec{E} &= \frac{1}{\varepsilon_0}\rho, \\ \nabla \times \vec{E} &= -\frac{\partial}{\partial t}\vec{B}, \\ \nabla \cdot \vec{B} &= 0, \\ c^2 \nabla \times \vec{B} &= \frac{1}{\varepsilon_0}\vec{j} + \frac{\partial}{\partial t}\vec{E}, \end{aligned}$$

   with ε_0 the dielectric constant of the medium and c^2 the speed of light in the medium. This defines the system $(\mathbb{R} \times \mathbb{R}^3, \mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R}, \mathfrak{B})$, with $\mathfrak{B}$ the set of all fields $(\vec{E}, \vec{B}, \vec{j}, \rho) : \mathbb{R} \times \mathbb{R}^3 \to \mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R}$ that satisfy Maxwell's equations.

3. *Kepler's laws* describe the possible motions of the planets in the solar system. This defines a dynamical system with $\mathbb{T} = \mathbb{R}$, $\mathbb{W} = \mathbb{R}^3$, and $\mathfrak{B}$ the set of maps $w : \mathbb{R} \to \mathbb{R}^3$ that satisfy Kepler's laws: the paths w must be ellipses in $\mathbb{R}^3$ with the sun (assumed in fixed position, say the origin of $\mathbb{R}^3$) in one of the foci; the radius vector from the sun to the planet must sweep out equal areas in equal time, and the ratio of the period of revolution around the ellipse to the major axis must be the same for all w's in $\mathfrak{B}$.

These examples fit perfectly our notion of a dynamical system as a triple $\Sigma = (\mathbb{T}, \mathbb{W}, \mathfrak{B})$ with $\mathfrak{B} \subseteq \mathbb{W}^{\mathbb{T}}$. Of course, the first two examples could be thought of as input/output systems. This already requires some goodwill in the case of Newton's second law in order to avoid a debate of causality in mechanics. But it is inappropriate to force Maxwell's equations (where there are also free variables in the system: the number of equations, 8, being strictly less than the number of variables, 10) into an input/output setting.

First principles laws in physics always state that some outcomes can happen (those satisfying the model equations) while others cannot happen (those violating the model equations). This is a far distance from specifying a system as being driven by free inputs which together with an initial state (whatever that is meant to be) specifies the other variables, the outputs. The behavioral framework treats a model for what it is: an exclusion law.

3 Interconnections and latent variables

Systems, especially engineering systems, usually consist of interconnections of subsystems. This feature is crucial in both modeling and design. The aim of this section is to formalize interconnections

and to analyze the model structures that emerge from it. We assume throughout finiteness, i.e., we assume that we interconnect a finite number of systems, each with a finite number of terminals, etc.

The building blocks of an interconnected system are *systems with terminals*. Each of these terminals carries variables from a universum, and the laws that governs the system are expressed by a behavior that relates these variables. Formally, a system Σ with T terminals has a behavior $\mathfrak{B} \subset \mathbb{U} = \mathbb{U}_1 \times \mathbb{U}_2 \times \cdots \times \mathbb{U}_T$. If $(u_1, u_2, \cdots, u_T) \in \mathfrak{B}$, then we think of $u_k \in \mathbb{U}_k$ as the variables realized at the k-th terminal.

As an example, consider an electrical component. We view this as an device that can interact with its environment through wires. These wires are the terminals. With each terminal we associate two real variables, the potential V and the current I (agreed to be positive when electrical current flows into the device). The laws of the device specify the behavior which will thus be a subset $\mathfrak{B}$ of the universum $\mathbb{R}^2 \times \mathbb{R}^2 \times \cdots \times \mathbb{R}^2 = (\mathbb{R}^2)^T$, where T denotes the number of terminal wires. Usually, the behavior $\mathfrak{B}$ will have to satisfy certain restrictions in order for it to qualify as the behavior of an electrical device. For example, *Kirchhoff's current law* and *Kirchhoff's voltage law*. These can be expressed as stating that $((V_1, I_1), \cdots, (V_T, I_T)) \in \mathfrak{B}$ must imply $I_1 + I_2 + \cdots + I_T = 0$ and $((V_1 + \alpha, I_1), \cdots, (V_T + \alpha, I_T)) \in \mathfrak{B}$ for all $\alpha \in \mathbb{R}$. There may be other requirements, as passivity, etc., but these will not concern us here.

For a thermal terminal, the terminal variables are the heat flow and the temperature. For a mechanical system, the terminal variables are position, and attitude, force and momentum (but it is much more involved to formalize interconnection in this case).

An interconnected system is specified by these subsystems, its building blocks, and by an interconnection architectures. The notion of a *graph with leaves* appears to be the appropriate concept for formalizing an interconnection architecture.

A *graph with leaves* is defined by 3 (disjoint) sets $(\mathbb{N}, \mathbb{E}, \mathbb{L})$, and two maps $(e, \ell); e : \mathbb{E} \to \tilde{\mathbb{N}^2}$, ($\tilde{\mathbb{N}^2}$ denotes the set of unordered pairs $\{n', n''\}$ with $n', n'' \in \mathbb{N}$) and $\ell : \mathbb{L} \to \mathbb{N}$. The set of $\mathbb{N}$ consists of the *nodes*, $\mathbb{E}$ of the *edges*, $\mathbb{L}$ of the *leaves*; if $e(\alpha) = \{n', n''\}$, then the edge α *connects* the nodes n' and n''; if $\ell(\beta) = n$, then the leave β is *attached* to the node n.

In associating a graph with leaves with an interconnection architecture, the nodes correspond to subsystems with terminals. These are the building blocks that are being connected. Edges that are connected to specific node and leaves that are attached to it, correspond to the terminals of the subsystem in that node. An edge signifies that the corresponding terminal of one subsystem is connected to the corresponding terminal of another (or in the case of a loop, that two terminals of the same system are connected). The leaves signify that the attached terminal is not connected and that it therefore serves as a terminal for the interconnected system.

It is assumed that by interconnecting two terminals by means of an edge, one imposes a restriction on the variables associated with these terminals. For example, if terminal t_1 with variables u_{t_1} is connected by an edge with terminal t_2 with variables u_{t_2}, we assume that a restriction is imposed on the pair (u_{t_1}, u_{t_2}). For instance, if t_1 and t_2 are both electrical terminals, this restriction is $V_{t_1} = V_{t_2}, I_{t_1} + I_{t_2} = 0$. If they are thermal terminals, this restriction is $q_{t_1} + q_{t_2} = 0$ (the heat flows are opposite) and $T_{t_1} = T_{t_2}$ (the temperatures are equal). Similar, but more complex, interconnection constraints can be formulated for mechanical connections, etc.

In an interconnection architecture there will usually also be the constraint that edges can only connect terminals that are of the same type (both electrical, both thermal, both mechanical, etc.). Also, a typical system that serves as a building block will have terminals of different type (a motor has electrical and mechanical terminals). However, we do not pursue these ideas here.

The behavior defined by an interconnected system is specified as follows. Its universum equals $\mathbb{U} = \mathbb{U}_{\ell_1} \times \cdots \times \mathbb{U}_{\ell_L}$, where $\mathbb{L} = (\ell_1, \cdots, \ell_L)$ is the set of leaves. The behavior is specified by the behavior of system in the nodes and by the edges. The variables on the terminals connected to a node and the leaves attached to it, must satisfy the laws of the subsystem associated with that node. The variables on the terminals of an edge must satisfy also the interconnection law resulting from the connection.

The resulting behavior $\mathfrak{B} \subset \mathbb{U}_{\ell_1} \times \cdots \times \mathbb{U}_{\ell_L}$ of the interconnected system is therefore specified in terms of the behaviors $\mathfrak{B}_{n_1}, \cdots, \mathfrak{B}_{n_N}$ of the system in the nodes, and the interconnection constraints. The important thing is that the specification of $\mathfrak{B}$ involves not only the variables on the leaves, but also those on the edges.

This presence of auxiliary variables in a model is basically an invariant of a first principles modeling procedure: in such a model there will essentially always be auxiliary variables involved in order to specify the laws of the system. It is therefore important to incorporate these auxiliary variables *ab initio* in a modeling framework. This leads to the notion of a model with *manifest* variables (the variables that

a model aims at) and *latent* variables (variables that have been introduced in the modeling process). Hence, a *mathematical model with latent variables* is defined as a triple $(\mathbb{U}, \mathbb{L}, \mathfrak{B}_{\text{full}})$ with $\mathbb{U}$ the universum of manifest variables, $\mathbb{L}$ the universum of latent variables, and $\mathfrak{B}_{\text{full}} \subseteq \mathbb{U} \times \mathbb{L}$ the full behavior. It induces the *manifest systems* $(\mathbb{U}, \mathfrak{B})$, with $\mathfrak{B} = \{u \in \mathbb{U} | \exists \ell \in \mathbb{L} \text{ such that } (u, \ell) \in \mathfrak{B}_{\text{full}}\}$.

A *dynamical system with latent variables* is defined completely analogously as $(\mathbb{T}, \mathbb{W}, \mathbb{L}, \mathfrak{B}_{\text{full}})$ with $\mathfrak{B}_{\text{full}} \subseteq (\mathbb{W} \times \mathbb{L})^{\mathbb{T}}$. The notion of a dynamical system with latent variables is the natural end-point of a modeling process and hence a very natural starting point for the analysis and synthesis of systems. We shall see that latent variables also enter very forcefully in representation questions.

Interconnected systems provide the prime example of the usefulness of behaviors and the inadequacy of input/output thinking. Even if our system, after interconnection, allows for a natural input/output representation, this is unlikely be the case of the subsystem and of the interconnection architecture. If the field of systems and control wants to take modeling seriously, is should retrace the *faux pas* of input/output thinking and cast models in the language of behaviors.

4 Differential systems

The 'ideology' that underlies the behavioral approach is the belief that in a model of a dynamical (physical) phenomenon, it is the behavior $\mathfrak{B}$, i.e., a set of trajectories $w : \mathbb{T} \to \mathbb{W}$, that is the central object of study. However, as we have seen, in first principles modeling, also latent variables enter *ab initio*. But, the set $\mathfrak{B}$ or $\mathfrak{B}_{\text{full}}$ of trajectories must be specified somehow, and it is here that differential (and difference) equations enter the scene. Of course, there are important examples where the behavior is specified in other ways (for example, in Kepler's laws for planetary motion), but differential equations are certainly the most prevalent specification of behaviors encountered in applications. For $\mathbb{T} = \mathbb{R}$, and in the case without latent variables, $\mathfrak{B}$ then consists of the solutions of a system of differential equations as

$$f_1(w, \frac{d}{dt}w, \dots, \frac{d^N}{dt^N}w) = f_2(w, \frac{d}{dt}w, \dots, \frac{d^N}{dt^N}w).$$

We call these *differential* systems. In the case of systems with latent variables these differential equation involves both manifest and latent variables. For $\mathbb{T} = \mathbb{R}^n$, this leads to partial differential equations.

Of particular interest (at least in control, signal processing, circuit theory, etc.) are systems with a signal space that is a finite-dimensional vector space and behavior described by linear constant-coefficient differential equations. A *1-D linear time-invariant differential system* is a dynamical system $\Sigma = (\mathbb{R}, \mathbb{W}, \mathfrak{B})$, with $\mathbb{W}$ a finite-dimensional (real) vector space, whose behavior consists of the solutions of

$$R(\frac{d}{dt})w = 0,$$

with $R \in \mathbb{R}^{\bullet \times \bullet}[\xi]$ a real polynomial matrix. We call this a *kernel representation* of the associated linear time-invariant differential system. Of course, the number of columns of R equals the dimension of $\mathbb{W}$. The number of rows of R, which represents the number of equations, is arbitrary. In fact, when the row dimension of R is less than its column dimension, as is usually the case, $R(\frac{d}{dt})w = 0$ is an under-determined system of differential equations which is typical for models in which the influence of the environment is taken into account. The precise definition of what we consider a solution of $R(\frac{d}{dt})w = 0$ is an issue that we will slide over.

The analogue for systems with latent variables, leads to

$$f_1(w, \frac{d}{dt}w, \dots, \frac{d^N}{dt^N}w, \ell, \frac{d}{dt}\ell, \dots, \frac{d^N}{dt^N}\ell) = f_2(w, \frac{d}{dt}w, \dots, \frac{d^N}{dt^N}w, \ell, \frac{d}{dt}\ell, \dots, \frac{d^N}{dt^N}\ell),$$

relating the (vector of) manifest variables w to the (vector of) latent variables ℓ. In the linear time-invariant case this becomes

$$R(\frac{d}{dt})w = M(\frac{d}{dt})\ell,$$

with R and M polynomial. Define the *manifest* behavior of this system as

$$\{w \mid \exists \ell \text{ such that } R(\frac{d}{dt})w = M(\frac{d}{dt})\ell\}.$$

We call the above differential equation involving ℓ a *latent variable* representation of the manifest behavior $\mathfrak{B}$. The question occurs whether $\mathfrak{B}$ can be described by a linear constant coefficient differential equation. This is the case indeed.

Theorem 1 : *For any real polynomial matrices (R, M) with* $\mathrm{rowdim}(R) = \mathrm{rowdim}(M)$, *there exists a real polynomial matrix R' such that the manifest behavior of $R(\frac{d}{dt})w = M(\frac{d}{dt})\ell$ has the kernel representation $R'(\frac{d}{dt})w = 0$.*

The above theorem is called the *elimination theorem*. Its relevance in object-oriented modeling is as follows. As we have seen for the simple electrical circuit discussed in the previous section, a model obtained this way usually involves very many variables and equations, among them many algebraic ones. The elimination theorem tells us that the latent variables may be eliminated and (in the case of linear time-invariant differential systems) that the number of equations can be reduced to no more than the number of manifest variables. Of course, the order of the differential equation goes up in the elimination process.

5 Controllability

An important property in the analysis and synthesis of dynamical systems is controllability. Controllability refers to be ability of transferring a system from one mode of operation to another. By viewing the first mode of operation as undesired and the second one as desirable, the relevance to control and other areas of applications becomes clear. The concept of controllability has originally been introduced in the context of state space systems. The classical definition runs as follows. The system described by the controlled vector-field $\frac{d}{dt}x = f(x, u)$ is said to be controllable if $\forall a, b, , \exists u$ and $T \geq 0$ such that the solution to $\frac{d}{dt}x = f(x, u)$ and $x(0) = a$ yields $x(T) = b$. One of the elementary results of system theory states that the finite-dimensional linear system $\frac{d}{dt}x = Ax + Bu$ is controllable if and only if the matrix $[B\ AB\ A^2B \cdots A^{\dim(x)-1}B]$ has full row rank. Various generalizations of this result to time-varying, to nonlinear (involving Lie brackets), and to infinite-dimensional systems exist.

A disadvantage of the notion of controllability as formulated above is that it refers to a particular representation of a system, notably a state space representation. Thus a system may be uncontrollable either for the intrinsic reason that the control has insufficient influence on the system variables, or because the state has been chosen in an inefficient way. It is clearly not desirable to confuse these reasons. In the context of behavioral systems, a definition of controllability has been put forward that involves the system variables directly.

Let $\Sigma = (\mathbb{T}, \mathbb{W}, \mathfrak{B})$ be a dynamical system with $\mathbb{T} = \mathbb{R}$ or $\mathbb{Z}$, and assume that is time-invariant, that is $\sigma^t\mathfrak{B} = \mathfrak{B}$ for all $t \in \mathbb{T}$, where σ^t denotes the t-shift (defined by $(\sigma^t f)(t') := f(t' + t)$); Σ is said to be *controllable* if for all $w_1, w_2 \in \mathfrak{B}$ there exists $T \in \mathbb{T}$, $T \geq 0$ and $w \in \mathfrak{B}$ such that $w(t) = w_1(t)$ for $t < 0$ and $w(t) = w_2(t - T)$ for $t \geq T$. Thus controllability refers to the ability to switch from any one trajectory in the behavior to any other one, allowing some time-delay.

Two questions that occur are the following: What conditions on the parameters of a system representation imply controllability? Do controllable systems admit a particular representation in which controllability becomes apparent? For linear time-invariant differential systems, these questions are answered in the following theorem.

Theorem 2 : *Let $\Sigma = (\mathbb{R}, \mathbb{R}^{\mathbf{w}}, \mathfrak{B})$ be a linear time-invariant differential system. The following are equivalent:*

1. *Σ is controllable;*
2. *The polynomial matrix R in a kernel representation $R(\frac{d}{dt})w = 0$ of $\mathfrak{B}$ satisfies* $\mathrm{rank}(R(\lambda)) = \mathrm{rank}(R)$ *for all $\lambda \in \mathbb{C}$;*
3. *The behavior $\mathfrak{B}$ is the image of a linear constant-coefficient differential operator, that is, there exists a polynomial matrix $M \in \mathbb{R}^{\mathbf{w}\times\bullet}[\xi]$ such that $\mathfrak{B} = \{w \mid w = M(\frac{d}{dt})\ell$ for some $\ell\}$.*

There exist various algorithms for verifying controllability of a system starting from the coefficients of the polynomial matrix R in a kernel (or a latent variable) representation of Σ, but we will not enter into these algorithmic aspects.

A point of the above theorem that is worth emphasizing is that, as stated in the above theorem, controllable systems admit a representation as the manifest behavior of the latent variable system of the special form

$$w = M(\frac{d}{dt})\ell.$$

We call this an *image* representation. It follows from the elimination theorem that every system in image representation can be brought in kernel representation. But not every system in kernel representation can be brought in image representation: it is precisely the controllable ones for which this is possible.

The controllability issue has been pursued for many other classes of systems. In particular (more difficult to prove) generalizations have been derived for differential-delay [10, 13], for nonlinear, for n-D systems [9, 11], and, as we will discuss soon, for PDE's. Systems in an image representation have received much attention recently for nonlinear differential-algebraic systems, where they are referred to as *flat* systems [14]. Flatness implies controllability, but the exact relation remains to be studied.

The controllability issue has been pursued for many other classes of systems. In particular (more difficult to prove) generalizations have been derived for differential-delay, nonlinear, and n-D systems, and, as we will discuss soon, for PDE's. Systems in an image representation have received much attention recently for nonlinear differential-algebraic systems, where they are referred to as *flat* systems. Flatness implies controllability, but the exact relation remains to be studied.

6 Observability

The notion of observability is always introduced hand in hand with controllability. In the context of the input/state/output system $\frac{d}{dt}x = f(x,u), y = h(x,u)$, it refers to the possibility of deducing, using the laws of the system, the state from observation of the input and the output. The definition that is used in the behavioral context is more general in that the variables that are observed and the variables that need to be deduced are kept general.

Let $\Sigma = (\mathbb{T}, \mathbb{W}, \mathfrak{B})$ be a dynamical system, and assume that $\mathbb{W}$ is a product space: $\mathbb{W} = \mathbb{W}_1 \times \mathbb{W}_2$. Then w_1 is said to be *observable* from w_2 in Σ if $(w_1, w_2') \in \mathfrak{B}$ and $(w_1, w_2'') \in \mathfrak{B}$ imply $w_2' = w_2''$. Observability thus refers to the possibility of deducing the trajectory w_1 from observation of w_2 and from the laws of the system ($\mathfrak{B}$ is assumed to be known).

The theory of observability runs parallel to that of controllability. We mention only the result that for linear time-invariant differential systems, w_1 is observable from w_2 if and only if there exists a set of differential equations satisfied by the behavior of the system of the following form that puts observability into evidence: $w_1 = R_2'(\frac{d}{dt})w_2$.

7 Distributed systems

We now explain the generalization of some of the above concepts and results to constant-coefficient PDE's. Define a *distributed differential system* as an n-D system $\Sigma = (\mathbb{R}^n, \mathbb{R}^w, \mathfrak{B})$, with behavior $\mathfrak{B}$ consisting of the solution set of a system of partial differential equations

$$R(\frac{\partial}{\partial x_1}, \dots, \frac{\partial}{\partial x_n})w = 0$$

viewed as an equation in the functions

$$(x_1, \dots, x_n) = x \in \mathbb{R}^n \mapsto (w_1(x), \dots, w_w(x)) = w(x) \in \mathbb{R}^w.$$

Here, $R \in \mathbb{R}^{\bullet \times w}[\xi_1, \dots, \xi_n]$ is a matrix of polynomials in $\mathbb{R}[\xi_1, \dots, \xi_n]$. Important properties of these systems are their *linearity* (meaning that $\mathfrak{B}$ is a linear subspace of $(\mathbb{R}^w)^{\mathbb{R}^n}$), and *shift-invariance* (meaning $\sigma^x\mathfrak{B} = \mathfrak{B}$ for all $x \in \mathbb{R}^n$, where σ^x denotes the $x-$shift, defined by $(\sigma^x f)(x') = f(x' + x)$). We call the above PDE a *kernel representation* of this $n - D$ system.

For distributed differential systems with latent variables, this leads to equations of the form

$$R(\frac{\partial}{\partial x_1}, \dots, \frac{\partial}{\partial x_n})w = M(\frac{\partial}{\partial x_1}, \dots, \frac{\partial}{\partial x_n})\ell,$$

with R and M matrices of polynomials in $\mathbb{R}[\xi_1, \ldots, \xi_n]$. This equation relates the (vector of) manifest variables w to the (vector of) latent variables ℓ. Define the *full behavior* of this system as

$$\mathfrak{B}_{\text{full}} = \{(w, \ell) \mid \text{ the PDE in } (w, \ell) \text{ holds}\}$$

and the *manifest behavior* as

$$\mathfrak{B} = \{w \mid \exists \ \ell \text{ such that } (w, \ell) \in \mathfrak{B}_{\text{full}}\}$$

We call the PDE with latent variables a *latent variable* representation of $\mathfrak{B}$. The question again occurs whether $\mathfrak{B}$ can itself be described by a set of $PDE's$. This is the case indeed.

Theorem 3 *For any pair of real matrices of polynomials (R, M) in $\mathbb{R}[\xi_1, \xi_2, \ldots, \xi_n]$ with* rowdim(R) = rowdim(M), *there exists a real matrix of polynomials R' in $\mathbb{R}[\xi_1, \xi_2, \ldots, \xi_n]$ such that the manifest behavior of $\mathfrak{B}$ has kernel representation $R'(\frac{\partial}{\partial x_1}, \ldots, \frac{\partial}{\partial x_n})w = 0$.*

As an illustration of the elimination theorem, consider the elimination of $\vec{B}$ and ρ from Maxwell's equations. The following equations describe the possible realizations of the fields $\vec{E}$ and $\vec{j}$:

$$\begin{aligned} \varepsilon_0 \frac{\partial}{\partial t} \nabla \cdot \vec{E} \ + \ \nabla \cdot \vec{j} &= \ 0, \\ \varepsilon_0 \frac{\partial^2}{\partial t^2} \vec{E} + \varepsilon_0 c^2 \nabla \times \nabla \times \vec{E} \ + \ \frac{\partial}{\partial t} \vec{j} &= \ 0. \end{aligned}$$

Note that it follows from the elimination theorem that the manifest behavior of a system in image representation, i.e., a latent variable system of the special form

$$w = M(\frac{\partial}{\partial x_1}, \ldots, \frac{\partial}{\partial x_n})\ell \tag{1}$$

can be described as the solution set of a system of constant coefficient $PDE's$. Whence, every image of a constant coefficient linear partial differential operator is the kernel of a constant coefficient linear partial differential operator. However, not every kernel of a constant coefficient linear partial differential operator is the image of a constant coefficient linear partial differential operator. The following theorem, obtained in [7], shows that it are precisely the controllable systems that admit an image representation.

Theorem 4 *The following statements are equivalent for systems described by constant coefficient linear PDE's:*

1. *$\mathfrak{B}$ defines a controllable system,*
2. *$\mathfrak{B}$ admits an image representation,*
3. *The trajectories of compact support are dense in $\mathfrak{B}$.*

It can be shown that Maxwell's equations define a controllable distributed differential system. Note that an image representation corresponds to what in mathematical physics is called a *potential function* with ℓ the potential and $M(\frac{\partial}{\partial x_1}, \ldots, \frac{\partial}{\partial x_n})$ the partial differential operator that generates elements of the behavior from the potential. An interesting aspect of the above theorem therefore is the fact that it identifies the existence of a potential function with the system theoretic property of controllability and concatenability of trajectories in the behavior. In the case of Maxwell's equations, an image representation is given by

$$\begin{aligned} \vec{E} &= -\frac{\partial}{\partial t}\vec{A} - \nabla\phi, \\ \vec{B} &= \nabla \times \vec{A}, \\ \vec{j} &= \varepsilon_0 \frac{\partial^2}{\partial t^2}\vec{A} - \varepsilon_0 c^2 \nabla^2 \vec{A}, \\ \rho &= \frac{\varepsilon_0}{c^2}\frac{\partial^2}{\partial t^2}\phi - \varepsilon_0 \nabla^2 \phi, \end{aligned}$$

where $\phi : \mathbb{R} \times \mathbb{R}^3 \to \mathbb{R}$ is a scalar, and $\vec{A} : \mathbb{R} \times \mathbb{R}^3 \to \mathbb{R}^3$ a vector potential. Note that Maxwell's equations consist of 8 equations in 10 variables. It turns out that the number of free variables is 3. In the above image representation there are 4 free latent variables. This can actually be reduced to 3, say by putting one component of $\vec{A}$ to zero. A more elegant way of reducing the freedom in the latent variables is by imposing a *gauge*, for example, restricting $\vec{A}$ and ϕ to satisfy $c^2 \nabla \cdot \vec{A} + \frac{\partial}{\partial t}\phi = 0$. Imposing this gauge retains the symmetry, but the resulting set of equations yields a latent variable representation of the behavior, not an image representation.

For distributed differential systems, w_1 is observable from w_2 if and only if there exists a set of annihilators of the behavior of the following form that puts observability into evidence: $w_1 = R_2'(\frac{\partial}{\partial x_1}, \ldots, \frac{\partial}{\partial x_n})w_2$, with $R_2' \in \mathbb{R}^{\dim(\mathfrak{w}_1) \times \dim(\mathfrak{w}_2)}[\xi_1, \ldots, \xi_n]$. We call a latent variable representation of the manifest behavior *observable* if ℓ is observable from w in its full behavior. We call it *weakly observable*, if to every $w \in \mathfrak{B}$ of compact support, there corresponds a unique ℓ that is also of compact support.

For *1-D* systems it is easy to show that every controllable linear time-invariant differential behavior $\mathfrak{B}$ admits an observable image representation. This, however, does not hold for *n-D* systems, and hence the representation of controllable systems in image representation (i.e., with potential functions) may require the introduction of latent variables that are 'hidden', in the sense that $M(\frac{\partial}{\partial x_1}, \ldots, \frac{\partial}{\partial x_n})\ell = 0$ has solutions $\ell \neq 0$. This means that however one represents a controllable behavior $\mathfrak{B}$ of a *PDE* as $w = M(\frac{\partial}{\partial x_1}, \ldots, \frac{\partial}{\partial x_n})\ell$, there may not exist an $N \in \mathbb{R}^{\mathfrak{w} \times \bullet}[\xi_1, \ldots, \xi_n]$ such that $w = M(\frac{\partial}{\partial x_1}, \ldots, \frac{\partial}{\partial x_n})\ell$ implies $\ell = N(\frac{\partial}{\partial x_1}, \ldots, \frac{\partial}{\partial x_n})w$. The latent variables do not be recoverable from the manifest ones by a 'local' differential operator. However, locally observable image representations always exist.

For example, the image representation of the behavior defined by Maxwell's equations in terms of the vector potential $\vec{A}$ and the scalar potential ϕ, is not observable (neither is the latent variable representation obtained after imposing the gauge, but then the resulting latent variable representation is weakly observable). In fact, Maxwell's equations are an example of a controllable system that does not allow an observable image representation.

8 Conclusions

In this paper, we have covered some highlights of the behavioral approach to systems and control. We view a mathematical model as a subset of an universum. However, in engineering applications, models are invariably obtained by interconnecting subsystems. This leads to the presence in mathematical models of manifest variables (the variables whose behavior the model aims at) and latent variables (the auxiliary variables introduced in the modeling process). Thus the central object in systems theory is a dynamical system with latent variables.

Various problems occur in this framework. For example, the elimination problem: obtaining differential equations for the manifest behavior that contain only the manifest variables. Further, the state space representation problem: obtaining a special latent variable representation in which the latent variables capture the memory of a system. There are many other representation questions, related to image representations, to input/output representations, etc.

In the behavioral framework, the concept of controllability becomes an intrinsic systems property related to concatenability of system trajectories. In the context of latent variable systems, observability refers to the possibility of deducing the latent variables in a system from observation of the manifest variables. In this way, these important concepts are extended far beyond the classical state space setting.

We view control as the design of a subsystem in an interconnected system, a subsystem that interacts with the plant through certain pre-specified variables, the control variables. For a linear time-invariant differential plant, it is possible to prove that a behavior is implementable by a linear time-invariant controller if and only if its behavior is wedged in between the hidden behavior and the realizable plant behavior.

The pre-occupation of systems and control with input/output systems does not do proper justice to the nature of physical systems: most physical systems are simply not a signal processors. Notwithstanding the importance of signal processors, the universal view of a system as an input/output device is simply a *faux pas*. And an unneccesary one at that: the behavioral approach offers a viable alternative.

References

[1] J.W. Polderman and J.C. Willems, *Introduction to Mathematical Systems Theory: A Behavioral Approach,* Springer-Verlag, 1998.

[2] J.C. Willems, System theoretic models for the analysis of physical systems, *Ricerche di Automatica,* Volume 10, pages 71-106, 1979.

[3] J.C. Willems, From time series to linear system - Part I. Finite dimensional linear time invariant systems, Part II. Exact modelling, Part III. Approximate modelling, *Automatica,* Volume 22, pages 561-580, 1986, Volume 22, pages 675-694, 1986, Volume 23, pages 87-115, 1987.

[4] J.C. Willems, Models for dynamics, *Dynamics Reported,* volume 2, pages 171-269, 1989.

[5] J.C. Willems, Paradigms and puzzles in the theory of dynamical systems, *IEEE Transactions on Automatic Control,* volume 36, pages 259-294, 1991.

[6] J.C.Willems, On interconnections, control, and feedback, *IEEE Transactions on Automatic Control,* volume 42, pages 326-339, 1997.

[7] H.K. Pillai and S. Shankar, A behavioral approach to control of distributed systems, *SIAM Journal on Control and Optimization,* volume 37, pages 388-408, 1999.

[8] J.C. Willems, Open dynamical systems and their control, Proceedings of the International Conference of Mathematicians, Berlin, *Documenta Mathematica,* Volume ICM 1998 - Invited papers, pages 697-706, 1998.

[9] P. Rocha and J.C. Willems, Controllability of 2-D systems, *IEEE Transactions on Automatic Control,* volume 36, pages 413-423, 1991.

[10] P. Rocha and J.C. Willems Behavioral controllability of delay-differential Systems, *SIAM Journal on Control and Optimization,* volume 35, pages 254-264, 1997.

[11] U. Oberst, Multidimensional constant linear systems, *Acta Applicandae Mathematicae,* volume 20, pages 1-175, 1990.

[12] S. Fröhler and U. Oberst, Continuous time-varying linear systems, *Systems & Control Letters,* volume 35, pages 97-110, 1998.

[13] H. Glüsing-Lüerssen, A behavioral approach to delay-differential systems, *SIAM Journal on Control and Optimization,* volume 35, pages 480-499, 1997.

[14] M. Fliess and S.T. Glad, An algebraic approach to linear and nonlinear control, pages 223-267 of *Essays on Control: Perspectives in the Theory and Its Applications,* edited by H.L. Trentelman and J.C. Willems, Birkhäuser, 1993.

[15] M. Fliess, J. Lévine, P. Martin, and P. Rouchon, Flatness and defect of nonlinear systems: introductory theory and applications, *International Journal on Control,* volume 61, pages 1327-1361, 1995.

THE INVERSE SIMULATION APPROACH : A FOCUSED REVIEW OF METHODS AND APPLICATIONS

D.J. Murray-Smith
Centre for Systems and Control and Department of Electronics and Electrical Engineering
University of Glasgow
Glasgow G12 8QQ, Scotland, U.K.

Abstract. Inverse techniques for dynamic simulation models which allow determination of the time history of "inputs" needed to achieve a specified time history for a selected set of "outputs" have been receiving some attention in recent years within aerospace engineering and in other application areas, including automatic control. This paper provides a review of some currently available methods and algorithms which have received particular attention in the field of aircraft flight mechanics and surveys a number of typical applications. The potential of the inverse simulation approach for external validation of simulation models is given special consideration.

Introduction

Using conventional simulation techniques a model "output" response is determined for a given set of initial conditions and a prescribed time history for a selected "input" variable. *Inverse simulation* may be defined as the reverse of that process, where the time history of a selected system "output" variable is prescribed and the inverse simulation algorithm allows the investigator to determine the time history of the corresponding "input" variable.

The inverse simulation approach is of practical value for a number of reasons and has attracted particular attention in the solution of nonlinear problems where interest is focused upon the control action needed to achieve a particular form of output response. It thus allows, for example, investigation of the characteristics needed in a control actuator to ensure that the overall system performance is not degraded due to amplitude or rate limits. Inverse simulation can also offer useful insight in manual control problems where interest is focused on the ability of the human operator to provide the control actions necessary to achieve a particular constrained response. The inverse approach can show very clearly when a particular task is likely to be beyond the capabilities of a human due to inherent dynamic limitations such as reaction time and neuromuscular lags.

Applications of inverse simulation have been particularly significant in aircraft flight control and in aircraft handling qualities investigations. For example, a mission goal may be specified in terms of the time history of the required vehicle trajectory and the simulated vehicle can be forced to fly that trajectory. Time histories of the state variables and control input variables for that mission can then be determined by inverse simulation and thus more fundamental design requirements in terms of forces, moments, torques and power output levels can be established. Similarly, the effects of configurational changes such as the mass or the position of the centre of gravity, can be investigated in a very straightforward fashion with results available in a form which allows direct and relatively easy interpretation. Insight concerning aircraft handling qualities and pilot workload is also directly available through the application of inverse simulation tools. This is one of the specialised areas in which particular progress has been made in recent years.

Another field in which inverse simulation techniques appear to have potential is in the external validation of nonlinear simulation models using time history data gathered from experiments on the corresponding real system. Comparison of the measurement input variables with equivalent variables from the simulation model when the simulation model is driven (in inverse mode) by the measured output response data from the real system can provide insight about model deficiencies which may not be so obvious from conventional output response comparisons.

Although inverse dynamics and inverse kinematics are topics which have received considerable attention from those working in the robotics, biomechanics and computer animation areas this review does not consider developments in these fields. The forms of mathematical description and the constraints used tend to

have specific forms in such application areas and these influence very strongly the approaches adopted. The main focus in this review is upon techniques which have been developed primarily for work in the field of aircraft flight mechanics modelling and have potential for applications in other areas involving nonlinear dynamic system models of a general kind.

The Inverse Mathematical Model

Dynamic system investigations normally involve solution of an initial value problem described by the equations

$$\dot{x} = f(x, u) \qquad x(o) = x_o \tag{1}$$

$$y = g(x) \tag{2}$$

where x is the system state vector, u is the control vector and y is the output vector. Thus in conventional simulation processes y(t) is found for a given u(t) and a given set of initial conditions x_o. Inverse simulation, on the other hand, involves calculation of the control input time history u(t) which is required in order to produce a given output time history y(t) [1].

Differentiating Eqn. (2) with respect to x gives

$$\dot{y} = \frac{dy}{dx}.\dot{x} = \frac{dg}{dx} f(x, u) \tag{3}$$

In simple cases where Eqn. (3) is invertible with respect to the variable u it is possible to create an equation of the form

$$u = h(x, \dot{y}) \tag{4}$$

and substitute this into Eqn. (1) to give

$$\dot{x} = f(x, h(x, \dot{y})) = F(x, \dot{y}) \tag{5}$$

Eqns. (4) and (5) thus provide a complete statement of the inverse model with $\dot{y}$ given and the vector u to be determined.

If Eqn. (3) is not invertible with respect to u further differentiation of Eqn. (2) is necessary to allow an inverse model to be obtained. This means that higher derivatives of y are then included as forcing terms in the inverse description.

It is important to note that the inverse model defined by Eqns. (4) and (5) has, in general, dynamic properties which differ significantly from the dynamics of the original model used for conventional simulation studies. Note also that the forcing function in Eqn. (5) is $\dot{y}(t)$ rather than y(t). This means that the form of the demanded output must be chosen with care. In particular, if y(t) is not a smooth function the derived input u(t) is unlikely to be of practical value. Although this may appear at first to be a restriction on the inverse method of approach it is actually a natural consequence of the dynamics of real physical systems. For example, discontinuities in a velocity time history will produce unrealistic accelerations and will be associated with forces which cannot be realised. In selecting a required output attention must be given to the range of attainable forces and other input variables.

Linearised Formulation

The linearised forms of Eqns. (1) and (2) can provide useful insight concerning the general problems of inverse modelling and simulation. Using a general linear description

$$\dot{x} = Ax + Bu \tag{6}$$

$$y = Cx \tag{7}$$

where A, B and C are the system, control and output matrices respectively, it can be shown [2] that the inverse problem has a unique solution provided

$$\dim(u) = \dim(y) = \text{rank}\begin{bmatrix} CB \\ CAB \\ \cdots \\ CA^{n-1}B \end{bmatrix} \tag{8}$$

It should be noted that the more items that are needed to establish the full rank then the higher the derivatives of y which are needed for calculation of the inverse solution [2].

Inverse Simulation Methods

A number of methods of inverse simulation have been developed and applied with success. Some of these methods have features which are specific to a particular area of application and have been developed for specialised problems. However, the approaches outlined in this paper are believed to be applicable in many different areas.

In general terms the available methods of inverse simulation may be divided into techniques which involve numerical differentiation and iterative techniques which are based upon numerical integration processes. Currently the methods based upon differentiation tend to be at least an order of magnitude faster than those which involve integration and the two approaches tend therefore to have different areas of application.

Approaches based upon numerical differentiation involve direct use of Eqns. (1) to (5) above. Eqns. (4) and (5) provide a complete statement of the inverse model for cases where Eqn. (3) is invertible with respect to u. In cases where this equation is not invertible higher derivatives of y must be included as forcing terms. Determination of $\dot{y}$ and higher derivatives of y requires the application of appropriate numerical differentiation techniques.

One differentiation-based approach which has been applied with considerable success to inverse problems in helicopter flight mechanics modelling [1] involves use of a simple implicit scheme for Eqn. (1). At time sample n Eqns. (1) and (2) may be re-written in discrete form as

$$\frac{x_n - x_{n-1}}{\Delta t} = f(x_n, u_n) \tag{6}$$

$$y_n = g(x_n) \tag{7}$$

where Δt is the time step between samples of variables u_n, x_n and y_n. The output vector y_n is known in inverse problems so the numerical procedure is based upon solution of Eqns. (6) and (7) for given y_n and x_{n-1}. In this approach the unknown values of x_n and u_n can be calculated using functions F_1 and F_2 defined as follows:

$$F_1(x_n, u_n) = f(x_n, u_n) - \frac{x_n - x_{n-1}}{\Delta t} \tag{8}$$

$$F_2(x_n, u_n = g(x_n) - y_n \tag{9}$$

The algorithm then has to determine values of x_n and u_n such that F_1 and F_2 are approximately zero. The Newton-Raphson method is appropriate for a problem of this kind. At the m^{th} iteration this gives

$$\begin{bmatrix} (x_n)_m \\ (u_n)_m \end{bmatrix} = \begin{bmatrix} (x_n)_{m-1} \\ (u_n)_{m-1} \end{bmatrix} - \begin{bmatrix} \frac{\partial F_1}{\partial x} & \frac{\partial F_1}{\partial u} \\ \frac{\partial F_2}{\partial x} & \frac{\partial F_2}{\partial u} \end{bmatrix}^{-1} \begin{bmatrix} F_1((x_n)_{m-1},(u_n)_{m-1}) \\ F_2((x_n)_{m-1}(u_n)_{m-1}) \end{bmatrix} \tag{10}$$

The iterative process is terminated when the elements of F_1 and F_2 reach predefined values close to zero. The corresponding values of x_n and u_n are the required quantities. The inverse simulation algorithm then continues the process to find the values of x and u at the next time step. By sequencing operations in a specific way significant computational benefits have been reported in the helicopter case [1] and similar benefits may be possible in other applications, although generalisation of the sequencing procedure has not so far proved possible.

Methods based upon numerical integration involve iterative procedures, with the most widely used integration-based approach being due to Hess, Gao and Wang [3]. This involves repeated solution of the initial value problem of Eqns. (1) and (2). The initial solution x_o can be determined in general from steady state calculations and the first estimate of the input u is then the value corresponding to that initial solution for x.

In the general case the m^{th} estimate at the n^{th} time point for the Hess, Gao and Wang method can be evaluated using the equations

$$(\dot{x}_{n-1})_m = f\left[(x_{n-1})_m,(u_{n-1})_m\right] \tag{11}$$

$$(x_n)_m = \int_{t_{n-1}}^{t_n} (\dot{x}_{n-1})_m \, dt + (x_{n-1})_m \tag{12}$$

$$(y_n)_m = g\left[(x_n)_m\right] \tag{13}$$

Then, by defining an error function e_n as the difference between $(y_n)_m$ and the required output $\bar{y}_n$ we have

$$(e_n)_m = (y_n)_m - \bar{y}_n \tag{14}$$

The elements of this error vector are compared with predefined threshold values. If the error exceeds the threshold a Newton-Raphson algorithm is used to obtain new estimates of the input as

$$(u_{n-1})_{m+1} = (u_{n-1})_m - [J]^{-1} (e_n)_m \tag{15}$$

where J is the Jacobian. New estimates of $\dot{x}_{n-1}$ and thus x_n and y_n are obtained and the iterative process continues with m being incremented until all elements of the error vector fall below the threshold values. The solution then continues to the next time point, incrementing m and n until all time points have been considered and the complete time history of the input variables has been calculated.

Limitations

Hess, Gao and Wang [3] categorise inverse simulation problems in three ways:-

- problems in which the number of inputs exceeds the number of outputs
- problems in which the number of inputs is equal to the number of outputs
- problems of the redundant type in which the number of inputs is smaller than the number of outputs

In general one can obtain solutions only for problems in the last two categories. Fortunately, practical situations in which the number of inputs is greater than the number of outputs seldom arise. In the redundant case the inverse Jacobian of Eqn. (15) does not exist and the Moore-Penrose generalised inverse has to be used.

The use of an algorithm based on numerical differencing such as in Eqns. (8) and (9) can give rise to problems of numerical rounding error. Problems can also be encountered if the functions to be differentiated in the calculation of elements of the Jacobian show low sensitivity to the unknown variables. The success of the algorithm based on Eqns. (8) and (9) is thus highly dependent upon the choice of the step size Δt and the increments used in the calculation of the elements of the Jacobian [4].

The choice of integration method and integration step size in the numerical methods which depend on integration techniques is of critical importance, as in any conventional simulation. Both accuracy and numerical stability issues arise and with a fixed-step integration method it is possible to investigate the problem systematically in a given application by observing results for successively smaller values of time step.

Gao and Hess [5] have discussed what they describe as "multiple solutions" and it has been suggested that this phenomenon occurs as a result of simulating a rapidly time-varying system with an integration step which is too large [4]. Problems have also been reported when very small time steps are employed [6].

Aircraft Applications

The calculation of the control inputs needed to allow an aircraft to fly a predefined manoeuvre is an interesting and potentially useful application of inverse simulation methods. Such an approach can be used to investigate the effect on a simulated aircraft of parametric changes at the design stage and can also provide valuable insight relating to issues of man-vehicle interaction such as aircraft handling qualities.

Two inverse simulation programs have been developed at the University of Glasgow which are specifically for helicopter applications. The first of these, HELINV, is of the numerical differentiation type [7] while the second, GENISA, involves an integration-based algorithm [4]. A simplified approach, known as NFPATH, has also been used by the helicopter manufacturer Agusta for preliminary design purposes [8] while for flight control system design other inverse-simulation-based techniques have been developed at the NASA Ames Research Center [9].

Comparative studies using the GENISA integration-based algorithm and the HELINV algorithm, based on numerical differentiation, show that it is possible to obtain results which are almost identical by the two methods. However the HELINV algorithm is significantly faster and is capable of providing faster-than-real-time results on currently available personal computers. One advantage found for the integration-based approach of GENISA is that the method can be made generic while HELINV is inherently model-specific. This means that with GENISA, and other integration-based software, any changes in the vehicle model can be introduced very easily while this is not the case with currently available methods which depend upon numerical differentiation. Results from the helicopter studies using HELINV and GENISA emphasise the importance of careful selection of the time increment Δt. In certain cases it has been found that both types of algorithm can show numerical instabilities unless Δt is chosen with great care [4].

In the helicopter field one of the most significant applications of inverse simulation has been in quantifying helicopter performance and in evaluation of handling qualities. In this type of situation the helicopter flight path becomes the input and the output of the inverse simulation program involves movements of the pilot's stick and pedals. Examples of investigations involving precision manoeuvring and helicopter handling qualities studies where inverse simulation techniques have proved beneficial may be found in the work of Bradley and Thomson [2], [10], [11], [12]. A further application involving an investigation of piloting strategies for engine failures during take-off from offshore platforms may be found in the work of Thomson,

Talbot et al. [13]. Other aircraft-related work which has been based upon inverse simulation methods includes the contributions of Kato and Siguira [14], Whalley [15] and McKillop and Perri [16].

Applications in External Validation of Models

The external validation of simulation models involving dynamic systems can be approached in a variety of ways [17]. Many commonly used techniques involve a comparison of the behaviour of variables within the simulation model with corresponding measured quantities in the real system. Inverse simulation methods offer interesting possibilities for model validation based on comparison of the input to the model and the system input needed to achieve a chosen output time history.

One potentially interesting application of inverse simulation techniques for external validation involves cases in which the dynamics of the system under investigation involve low frequency characteristics which are dominated by a simple integrator. This type of situation can arise in the validation of mathematical models of fixed wing aircraft and helicopters [18] as well as many other types of engineering system. In such situations any small offsets at the input to the system tend to produce an output which drifts steadily. If such offsets are not known and are incorporated in the model, conventional comparisons of the model and system outputs can prove very difficult. Inverse simulation avoids such problems and allows input offsets of this kind to be identified explicitly.

The application of inverse simulation methods to the external validation of a nonlinear model of a coupled tank process system [19] has provided some useful practical experience. It was found that because the input can have a very simple form, such as a step, model deficiencies which may be difficult to detect through forward simulation can show up very clearly in the inverse case. The results do not however point directly to the precise nature of the model imperfections in terms of structure or parameter values. Physical insight or further simulation-based investigations are needed to establish dependencies of the input estimated by the inverse simulation on the parameters and structure of the inverse model.

The coupled-tank study has shown that measurement noise can present difficulties when inverse simulation methods are used, even in the case of integration-based methods. Low-pass filtering of calculated inputs of the inverse modelling may be necessary in order to extract useful information.

One difficulty which limits the usefulness of inverse simulation techniques for routine application involving external validation problems is that the computational demands are very high. The coupled-tanks investigation showed that even with a relatively simple nonlinear model many minutes and possibly hours of central processor time on a modern personal computer could be needed for a single inverse simulation run using an integration-based algorithm. A detailed evaluation of possible model deficiencies using this inverse simulation approach would therefore be extremely time consuming.

Inverse simulation methods derived from an explicit model following control system technique developed within the Institute of Flight Mechanics of DLR in Germany have been applied successfully by Gray and von Grünhagen [20] to a problem of helicopter engine modelling. The methodology used involved a combination of an "open loop" simulation approach (in which experimental data is used to represent unknown states) with inverse simulation techniques. The work has been carried out in the context of the development of a nonlinear real-time helicopter simulation model.

Discussion and Conclusions

In the investigation of dynamic systems, particularly those described by nonlinear models, inverse simulation techniques can provide useful insight which complements the understanding which can come from more conventional modelling and simulation methods. Inverse simulation can be particularly useful as a development tool in the case of systems from which a very precisely defined form of output is required.

Although comparative applications studies show that the two main approaches to inverse simulation currently available provide very similar results and the inputs generated by inverse simulation are verifiable through tests based on forward simulation, it is known that problems of numerical instability can occur. The overall robustness of solutions can also present serious problems, especially in cases where time histories used

to drive an inverse simulation include measurement noise. The choice of time step to be used in inverse simulation can also present difficulties.

One advantage of the integration-based approach to inverse simulation is that it can be made generic while that based on differentiation is inherently model-specific. On the other hand the integration-based methods suffer from the disadvantage of being computationally intensive and thus, typically, an order of magnitude slower.

In using inverse simulation to determine the inputs which must be applied to ensure that a specified output time history is achieved it is important to define outputs which are of an appropriately smooth form, consistent with the dynamic characteristics of the system under investigation. It is also important for the user of inverse simulation methods to understand that the dynamics of an inverse model representing a system constrained at the output to provide a specified response can be very different in character from the dynamics of the equivalent unconstrained model when studied using conventional methods of simulation.

Engineering applications of inverse simulation have included initial system design and performance assessment studies, investigations of handling qualities and work load for pilots in fixed-wing aircraft and rotorcraft, the design of automatic and manual control systems and the external validation of dynamic system models. It is likely that this list will be extended very significantly in future.

Although well established as a flight-mechanics tool the inverse simulation methods described here have been used for relatively few other applications. It is also important to establish links with those working on inverse dynamics problems in areas such as robotics and biomechanics. The methods currently available are well documented but are not widely known within the mathematical modelling and simulation community. The tools available are far from perfect in their present form and the field of inverse simulation could well provide an interesting opportunity for new methodological developments.

Acknowledgements

I wish to acknowledge many helpful discussions with Dr. Douglas Thomson of the Department of Aerospace Engineering, University of Glasgow and Professor Roy Bradley of the Department of Mathematics, Glasgow Caledonian University who jointly pioneered the use within the University of Glasgow of inverse simulation methods. I also wish to acknowledge the contribution of some of my present and former students who have applied inverse methods to a variety of problems, especially Peter Galloway, Wong Boon Onn, Michael Baro and Neil Cameron.

References

[1] Thomson, D.G. and Bradley, R., The principles and practical application of helicopter inverse simulation. Simulation Practice and Theory, 6 (1998), 47-70.

[2] Bradley, R. and Thomson, D.G., Handling qualities and performance aspects of the simulation of helicopters flying mission task elements. In Proc. 18th European Rotorcraft Forum, Avignon, France, 1992, 139.1-139.15.

[3] Hess, R.A., Gao, C. and Wang, S.H., A generalized technique for inverse simulation applied to aircraft manoeuvres. J. Guidance, Control and Dynamics, 14 (1991), 920-926.

[4] Rutherford, S. and Thomson, D.G., Improved methodology for inverse simulation, Aeronautical J., 100 (1996), 79-86.

[5] Gao, C. and Hess, R.A., Inverse simulation of large-amplitude aircraft manoeuvres, J. Guidance, Control and Dynamics, 16 (1993), 733-737.

[6] Lin, K.C., Lu, P. and Smith, M., The numerical errors in inverse simulation, AIAA-93-3588-CP, 1993.

[7] Thomson, D.G. and Bradley, R., Development and verification of an algorithm for helicopter inverse simulation. Vertica, 14 (1990), 185-200.

[8] Nannoni, F. and Stabellini, A., A simplified inverse simulation for preliminary design purposes, In Proc. 15th European Rotorcraft Forum, Amsterdam, The Netherlands, 1989.

[9] Smith, G.A. and Meyer, G., Aircraft automatic control system with model inversion. J. Guidance and Control, 10 (1987).

[10] Thomson, D.G. and Bradley, R., Prediction of the dynamic characteristics of helicopters in constrained flight. Aeronautical J., 94 (1990), 344-354.

[11] Thomson, D.G. and Bradley, R., Modelling and classification of helicopter combat manoeuvres. In: Proc. 17th ICAS Congress, Stockholm, Sweden, 1990, 1763-1773.

[12] Thomson, D.G. and Bradley, R., The contribution of inverse simulation to the assessment of helicopter handling qualities. In: Proc. 19th ICAS Congress, Anaheim, U.S.A., 1994, 229-239.

[13] Thomson, D.G., Talbot, N., Taylor, C., Bradley, R. and Ablett, R., An investigation of piloting strategies for engine failures during take-off from offshore platforms. Aeronautical J., 99 (1995), 15-25.

[14] Kato, O. and Suguira, I., An interpretation of airplane general motion and control as an inverse problem. J. Guidance, Control and Dynamics, 9 (1986), 198-204.

[15] Whalley, M.S., Development and evaluation of an inverse solution technique for studying helicopter manoeuvrability and agility. NASA TM 102889, July 1991.

[16] McKillop, R.M. and Perri, T.A., Helicopter flight control system design and evaluation for NOE operations using controller inversion techniques, In Proceedings 45th Annual Forum of American Helicopter Society, Boston, U.S.A., May 1989.

[17] Murray-Smith, D.J., Methods for the external validation of continuous system simulation models: a review. Math. and Computer Modelling of Dynamical Systems, 4 (1998), 5-31.

[18] Bradley, R., Padfield, G.D., Murray-Smith, D.J. and Thomson, D.G., Validation of helicopter mathematical models. Trans. Inst. Measurement and Control, 12 (1990), 186-196.

[19] Murray-Smith, D.J. and Wong, B.O., Inverse simulation techniques applied to the external validation of nonlinear dynamic models. In: Proc. UK Sim '97, Third Conference of United Kingdom Simulation Society, 1997, 100-104.

APPLICATION OF COMPUTER ALGEBRA SIMULATION (CALS) IN INDUSTRY

S.Braun
Visual Analysis GmbH
Neumarkter Str. 87
D-81673 München
Tel: +49/89/431981-0
Fax: +49/89/431981-1
e-mail stefan.braun@visualanalysis.com
www.visualanalysis.com

Abstract

This talk treats the industrial application of Computeralgebra-Simulation (CALS). CALS is a combination of symbolic and numeric methods, which is very well suited for efficient solving of complex problems. Because it is an innovative 'hybrid' technique, completely new ways open up for approaching practical problems.
The basis of CALS are mathematical models reproducing reality in sufficient detail, so that CALS is independent of any specific field. Based on *Mathematica*, it is outlined how this method can be brought to bear on real-life problems.

Introduction

New and improved simulation techniques are in encreasing demand, because development cycles need to be shortened more and more. Traditional methods, based on a purely numerical approach, are already established in many fields. But it is not only the shortening of developmental periods, but also the growing complexity of product characteristics, which make a better understanding of process parameters necessary. Purely numerical simulation techniques are not really suited for that kind of problem: Even if they impress by giving detailed solutions of complicated structures and processes, most often they can model only a detail of the problem at hand - but they cannot yield information about the interaction and interdependence of the parameters. They only are efficient if all the physical parameters are known in sufficient accuracy; otherwise the efficiency is significantly reduced by having to calculate the same problem very often with slightly varying parameters.

The Computeralgebra-Simulation closes that gap by allowing a deeper insight into the physical functionality of technical processes.

Components of a Computeralgebrasystem

Symbolic Mathematics

A Computeralgebrasystem makes it possible to classify and interpret mathematical expressions. In other words, a Computeralgebrasystem can process mathematical expressions so that sophisticated manipulation of these expressions becomes possible. The symbolic capabilities cover all essential fields of Mathematics like solving equations and equation systems, calculating derivatives and integrals (calculus), solving differential equations,

etc.
For a better understanding, some examples are shown below:

Solving an equation:

$$\text{Solve}[x^3 + a == 1, x]$$

$$\{\{x \to \sqrt[3]{1-a}\}, \{x \to -\sqrt[3]{-1}\ \sqrt[3]{1-a}\}, \{x \to (-1)^{2/3}\ \sqrt[3]{1-a}\}\}$$

Calculating a derivative:

$$\frac{\partial J_n(x)}{\partial x}$$

$$\frac{1}{2}\left(J_{n-1}(x) - J_{n+1}(x)\right)$$

(Note that most Computeralgebrasystems have implemented a wide range of higher functions. So now mathematical tables aren't necessary any more, the values for specific variables can easily be calculated; also many more integrals, differential equations etc. involving higher functions become solvable.)

Calculating an integral:

$$\int_a^b \sin(x)\, e^{-x^2}\, dx$$

$$\frac{i\sqrt{\pi}\left(\text{erf}\left(\frac{1}{2}(2a - i)\right) - \text{erf}\left(\frac{1}{2}(2a + i)\right)\right)}{4\sqrt[4]{e}} - \frac{i\sqrt{\pi}\left(\text{erf}\left(\frac{1}{2}(2b - i)\right) - \text{erf}\left(\frac{1}{2}(2b + i)\right)\right)}{4\sqrt[4]{e}}$$

Solving a differential equation:

$$\text{DSolve}[x\, y(x) + y'(x) == \sin(x), y(x), x]$$

$$\left\{\left\{y(x) \to e^{-\frac{x^2}{2}} c_1 + \frac{1}{2} i e^{\frac{1}{2} - \frac{x^2}{2}} \sqrt{\frac{\pi}{2}} \left(\text{erfi}\left(\frac{x - i}{\sqrt{2}}\right) - \text{erfi}\left(\frac{x + i}{\sqrt{2}}\right)\right)\right\}\right\}$$

Numerical Mathematics

With a Computeralgebrasystem numeric calculations of arbitrary precision can be done. This capability also covers all important mathematical fields, e.g. equations, integrals, differential equations, nonlinear fitting, etc. By also using symbolic methods for numerical calculations, methods for automated decision making routines can be applied to numerical algorithms. An example for this is the numerical solving of equations (Newton's method) or differential equations (calculation of eigenvalues of the Jacobi-matrix for deciding if the equation is stiff or not).

Solving equations:

$$\text{FindRoot}[e^x == x^2, \{x, -0.5\}]$$

$$\{x \to -0.703467\}$$

Calculating integrals:

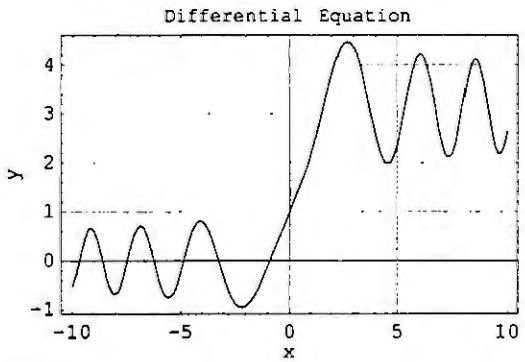
$$\text{NIntegrate}\left[\sin(x^2)\, e^{-x^2},\ \{x,\ 1,\ 2\}\right]$$

0.109562

Solving differential equations:

$$\text{lsg} = \text{NDSolve}[\{y''(x) == x \sin(y(x)),\ y(0) == 1,\ y'(0) == 1\},\ y(x),\ \{x,\ -10,\ 10\}];$$

$$\text{Plot}[\text{Evaluate}[y(x) /.\ \text{lsg}],\ \{x,\ -10,\ 10\},\ \text{FrameLabel} \to \{\text{"x"},\ \text{"y"}\},\ \text{PlotLabel} \to \text{"Differential Equation"},\ \text{GridLines} \to \text{Automatic},\ \text{Frame} \to \text{True}]$$



- Graphics -

Nonlinear Fits

```
<< Statistics`NonlinearFit`
```

$$\text{data} = \begin{pmatrix} 0 & 0.0195 \\ 0.3 & 0.61 \\ 1. & 1.34 \\ 1.5 & 1.74 \\ 2. & 2 \\ 2.29 & 1.9 \end{pmatrix};$$

$$\text{NonlinearFit}\left(\text{data},\ a + d\, e^{-e\,(f-x)^2} - b\, e^{-c\,x},\ \{x\},\ \{a,\ b,\ c,\ d,\ e,\ f\}\right)$$

$$1.75031 + 0.367313\, e^{-3.49469\,(1.90924-x)^2} - 1.73081\, e^{-1.39085\,x}$$

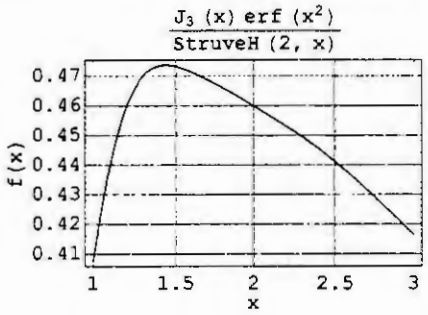
Visualization of Formulae and Data

With a Computeralgebrasystem either mathematical expressions or data or both can be visualized. This can be done in 2 or 3 dimensions.

2-dimensional plots

$$\text{Plot}\left[\frac{J_3(x)\,\text{erf}(x^2)}{\text{StruveH}(2,\,x)},\ \{x,\ 1,\ 3\},\ \text{PlotRange} \to \text{All},\ \text{GridLines} \to \text{Automatic},\right.$$

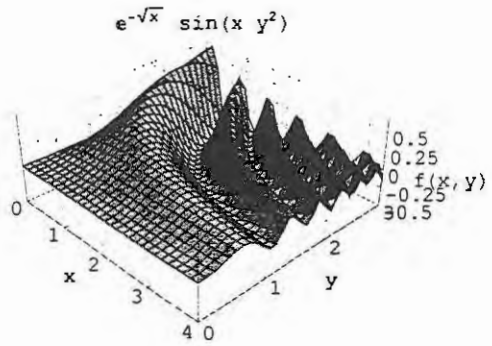
$$\left.\text{Frame} \to \text{True},\ \text{PlotLabel} \to "\frac{J_3\,(x)\backslash\ \text{erf}\,(x^2)}{\text{StruveH}\,(2,\ \backslash\ x)}",\ \text{FrameLabel} \to \{"x",\ "f(x)"\}\right]$$



- Graphics -

3-dimensional plots:

$$\text{Plot3D}\left[e^{-\sqrt{x}}\ \sin(x\,y^2),\ \{x,\ 0,\ 4\},\ \{y,\ 0,\ 3\},\ \text{PlotLabel} \to "e^{-\sqrt{x}}\ \sin(x\,y^2)",\right.$$
$$\text{FaceGrids} \to \text{All},\ \text{PlotPoints} \to 30,\ \text{ViewPoint} \to \{2.084,\ -1.988,\ 1.775\},$$
$$\left.\text{PlotRange} \to \text{All},\ \text{AxesLabel} \to \{"x",\ "y",\ "f(x,y)"\}\right]$$



- SurfaceGraphics -

Programming

By the help of the so-called 'Pattern Matching' method, completely new ways of programming become possible in a Computeralgebrasystem. Also provided are list-oriented methods and all elements known from conventional programming languages.

Pattern matching applied to a formula:

$$(e^x)^4 + x^3 + \sin^{2.5}(x) + x^2 \text{ /. } \{x\_^{\,2} \to u,\ x\_^{\,2.5} \to v\}$$

$$x^3 + e^{4x} + u + v$$

As you see in this case, all expressions involving a ***^2***, represented by the pattern object ***x_^2***, are replaced by ***u***. The same is valid for expressions involving ***^2.5***, which are replaced by ***v***.
This kind of method allows to treat expressions in a very efficient way, offering many possibilities of programming in a short an comprehensive style.

Applying a function to a list:

$$\sin\left(\begin{pmatrix} 1 & 2 & a \\ d & e & f \\ 1 & 2 & 3 \end{pmatrix}\right)$$

$$\begin{pmatrix} \sin(1) & \sin(2) & \sin(a) \\ \sin(d) & \sin(e) & \sin(f) \\ \sin(1) & \sin(2) & \sin(3) \end{pmatrix}$$

Programming a FOR-loop:

$$\text{For}\left[i = 1,\ i < 3,\ i{+}{+},\ \text{Print}\left[\frac{\partial J_i(x)}{\partial x}\right]\right]$$

$$\frac{1}{2}\,(J_0(x) - J_2(x))$$

$$\frac{1}{2}\,(J_1(x) - J_3(x))$$

Definition of Computeralgebra Simulation (CALS)

The combination of the basic elements of a Computeralgeabrasystem and its application for solving industrial problems is called *Computeralgebra-Simulation* (CALS).

Differences and advantages compared to other simulation techniques

There are many differences between CALS and other simulation techniques (e.g. numeric): The Computeralgebra-Simulation is different because of its combination of symbolic and numeric methods and the resulting possibilities for treading new ground in solution techniques.
Examples for purely numerical methods are FEM-Simulation (Nastran, Ansys, Marc) or traditional numeric simulation (Matlab, MatrixX, ACSL).

In the following few lines are listed some of the most important differences between CALS and the other methods:
Not every parameter needs to be given as numerical data
Exact solution in contrast to numerical approximation
Parametric models
Inverse problems are easy to solve
High flexibility
New kind of solution methods
Optimization becomes possible because of closed expressions
Intuitive and engineering style of working

Application areas where CALS is useful

The application areas of Computeralgebra-Simulation cover the complete area where Mathematics can be used in industry. A few examples of fields are
Air Conditioning and Heat Technology, Process Technology, Mechatronics, Control Systems, Vacuum Technology,
Mechatronics, Exhaust Gas Technology, Sensor Technology, Vibration Analysis, Micro System Technology, Fluid Mechanics

Optimization of a Coolfin

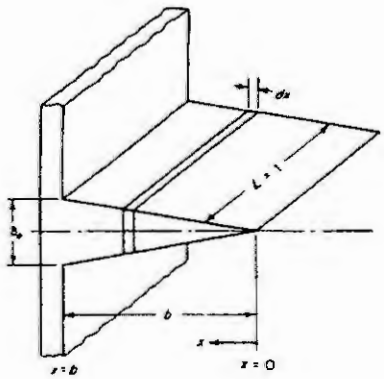


Figure 1.1: cooling fin

Today, electric components need an increasing amount of power, so that it becomes more and more important to find an optimized design for the neccessary cooling fins that dissipate the heat energy. A cooling fin is optimal, when it transports the most heat energy with constant material demand. This example demonstrates how Computeralgebra-Simulation can be used for solving this particular problem.

The following assumptions are made before the temperature distribution of the cooling fin is calculated:

1. The fin is so thin, that the temperature is dependent only on the coordinate in direction of the basis of the fin to its tip.
2. The material of the fin is homogeneous and has a constant thermal conductivity λ
3. The heat transmission on the surface of the tip is described by the constant heat transfer coeffiecient α

4. The temperature of the fluid surrounding the fin is constant
5. The heat flow at the tip of the fin is neglected with regard to the heat flow at the side areas

The differential equation for a profile of the fin, where the profile is discribed by y = y(x) is:

$$-\frac{\alpha\, T(x)}{\lambda} + \frac{T'(x)\, y'(x)}{y(x)} + T''(x) == 0$$

The boundary conditions for this differential equation are: T[0] == T0 - Tu and T'[h] == 0

The solution of the differential equation is done in *Mathematica* by the help of the add-on *Industrial Thermics*. The function for this is called *CoolFin*.

```
Needs["indtherm`indtherm`"]
```

Calculation of the temperatures

The output of the function CoolFin is the temperature distribution in a triangular cooling fin, if you set the option "Geometry → Triangle".

```
Temperature = CoolFin(λ, α, b, H, L, {T0, TF}, x, Geometry → Triangle)
```

$$\frac{(T0 - TF)\, J_0\left(2\sqrt{2}\,\sqrt{H(x-H)}\,\sqrt{\frac{\alpha}{b\lambda}}\right)}{J_0\left(2\sqrt{2}\,\sqrt{-H^2}\,\sqrt{\frac{\alpha}{b\lambda}}\right)}$$

Calculation of the heat flow

The heat flow dissipated by the cooling fin is defined by:

$$q = -\lambda\, a\, L\, \frac{\partial T(x)}{\partial x} \Big/ x \to 0$$

If you set the options "Geometry → Triangle" and "Result → HeatFlow", the output of the function CoolFin will be the heat flow in the triangular cooling fin.

```
Heatflow =
  PowerExpand[CoolFin(λ, α, b, H, L, {T0, TF}, x, Geometry → Triangle, Result → HeatFlow)]
```

$$-\frac{i\sqrt{2}\,\sqrt{b}\,L\,(T0 - TF)\,\sqrt{\alpha}\,\sqrt{\lambda}\,J_1\left(\frac{2i\sqrt{2}\,H\sqrt{\alpha}}{\sqrt{b}\,\sqrt{\lambda}}\right)}{J_0\left(\frac{2i\sqrt{2}\,H\sqrt{\alpha}}{\sqrt{b}\,\sqrt{\lambda}}\right)}$$

Optimization of the cooling fin

Defining the volume of the cooling fin and solving for the width of the fin:

$$\text{Vsol2} = \text{Solve}\left[V == \frac{H\,b\,L}{2},\ b\right]$$

$$\left\{\left\{b \to \frac{2\,V}{H\,L}\right\}\right\}$$

Putting the result in the equation for the heat flow:

$$\text{Heatflow} = \text{Heatflow} /.\ \text{Vsol2}[[1]]$$

$$-\frac{2\,i\,L\,(\text{T0}-\text{TF})\sqrt{\frac{V}{HL}}\,\sqrt{\alpha}\,\sqrt{\lambda}\,J_1\left(\frac{2\,i\,H\sqrt{\alpha}}{\sqrt{\frac{V}{HL}}\,\sqrt{\lambda}}\right)}{J_0\left(\frac{2\,i\,H\sqrt{\alpha}}{\sqrt{\frac{V}{HL}}\,\sqrt{\lambda}}\right)}$$

For maximal heatflow, the derivative of the above equation is zero:

$$\text{Simplify}\left[\frac{\frac{\partial \text{Heatflow}}{\partial H}}{\text{T0}-\text{TF}}\right]$$

$$\frac{1}{H\,J_0\left(\frac{2\,i\,H\sqrt{\alpha}}{\sqrt{\frac{V}{HL}}\,\sqrt{\lambda}}\right)^2}\left(L\left(3\,H\,\alpha\,J_0\left(\frac{2\,i\,H\sqrt{\alpha}}{\sqrt{\frac{V}{HL}}\,\sqrt{\lambda}}\right)^2+\left(i\sqrt{\frac{V}{HL}}\,\sqrt{\alpha}\,\sqrt{\lambda}\,J_1\left(\frac{2\,i\,H\sqrt{\alpha}}{\sqrt{\frac{V}{HL}}\,\sqrt{\lambda}}\right)-3\,H\,\alpha\,J_2\left(\frac{2\,i\,H\sqrt{\alpha}}{\sqrt{\frac{V}{HL}}\,\sqrt{\lambda}}\right)\right)J_0\left(\frac{2\,i\,H\sqrt{\alpha}}{\sqrt{\frac{V}{HL}}\,\sqrt{\lambda}}\right)+6\,H\,\alpha\,J_1\left(\frac{2\,i\,H\sqrt{\alpha}}{\sqrt{\frac{V}{HL}}\,\sqrt{\lambda}}\right)^2\right)\right)$$

Simplifying this equation yields:

$$\text{EquationHeight} = \text{Expand}\left[\text{Simplify}\left[\text{PowerExpand}\left[\frac{\%}{L\,\alpha}\right]\right]\right]$$

$$\frac{6\,J_1\left(\frac{2\,i\,H^{3/2}\sqrt{L}\,\sqrt{\alpha}}{\sqrt{V}\,\sqrt{\lambda}}\right)^2}{J_0\left(\frac{2\,i\,H^{3/2}\sqrt{L}\,\sqrt{\alpha}}{\sqrt{V}\,\sqrt{\lambda}}\right)^2}+\frac{i\sqrt{V}\,\sqrt{\lambda}\,J_1\left(\frac{2\,i\,H^{3/2}\sqrt{L}\,\sqrt{\alpha}}{\sqrt{V}\,\sqrt{\lambda}}\right)}{H^{3/2}\sqrt{L}\,\sqrt{\alpha}\,J_0\left(\frac{2\,i\,H^{3/2}\sqrt{L}\,\sqrt{\alpha}}{\sqrt{V}\,\sqrt{\lambda}}\right)}-\frac{3\,J_2\left(\frac{2\,i\,H^{3/2}\sqrt{L}\,\sqrt{\alpha}}{\sqrt{V}\,\sqrt{\lambda}}\right)}{J_0\left(\frac{2\,i\,H^{3/2}\sqrt{L}\,\sqrt{\alpha}}{\sqrt{V}\,\sqrt{\lambda}}\right)}+3$$

The root of this equation cannot be calculated by using symbolic methods. But if you substitute a new variable and scale the equation by it, the equation can be solved numerically for the transformed variable. Then an inverse transformation can done for the equation.

Pattern matching for the transformation, where arguments of the Bessel functions are replaced by ***I*m***. The arguments of all the Bessel functions are the same $\left(\frac{2\,i\,H^{3/2}\sqrt{L}\,\sqrt{\alpha}}{\sqrt{V}\,\sqrt{\lambda}}\right)$, so that the result is:

EquationHeight = EquationHeight /. $\left\{J_{n\_}(x\_) :\to J_n\left(i\left(\text{ak} = \frac{x}{i}; m\right)\right)\right\}$

$$\frac{6\,J_1(i\,m)^2}{J_0(i\,m)^2} + \frac{i\sqrt{V}\,\sqrt{\lambda}\,J_1(i\,m)}{H^{3/2}\sqrt{L}\,\sqrt{\alpha}\,J_0(i\,m)} - \frac{3\,J_2(i\,m)}{J_0(i\,m)} + 3$$

The argument $\left(\frac{2\,i\,H^{3/2}\sqrt{L}\,\sqrt{\alpha}}{\sqrt{V}\,\sqrt{\lambda}}\right)$ also appears outside the Bessel functions in the equation itself and is also replaced:

EquationHeight = $\left(\text{EquationHeight} /. \left\{\frac{i\sqrt{V}\,\sqrt{\lambda}}{H^{3/2}\sqrt{L}\,\sqrt{\alpha}} \to \frac{i\,2}{m}\right\}\right) == 0$

$$\frac{6\,J_1(i\,m)^2}{J_0(i\,m)^2} + \frac{2\,i\,J_1(i\,m)}{m\,J_0(i\,m)} - \frac{3\,J_2(i\,m)}{J_0(i\,m)} + 3 == 0$$

The remaining equation can be solved for ***m***:

Rootm = FindRoot[EquationHeight, {*m*, 1}]

$\{m \to 2.6188 + 0.\,i\}$

The solution for ***m*** is used to evaluate the necessary height H of the fin:

PowerExpand$\left[\text{ak} /. V \to \frac{H\,b\,L}{2}\right] == m /.$ Rootm[[1]]

$$\frac{2\sqrt{2}\,H\sqrt{\alpha}}{\sqrt{b}\,\sqrt{\lambda}} == 2.6188 + 0.\,i$$

OptHeight = Solve[%, *H*]

$$\left\{\left\{H \to \frac{(0.925887 + 0.\,i)\sqrt{b}\,\sqrt{\lambda}}{\sqrt{\alpha}}\right\}\right\}$$

References

1. Baehr, H. D. and Stephan, K., Wärme- und Stoffübertragung, Springer-Verlag Berlin Heidelberg, 1994

2. Braun, S. and Häuser, H., Computeralgebra im industriellen Einsatz - ein konkretes Problem, In: Spektrum der Wissenschaft, März 1996, 93 - 95

3. Carslaw, H. S. and Jaeger, J. C., Conduction of Heat in Solids, 2nd ed., Oxford University Press 1995

MODELLING FOR FAULT DETECTION AND ISOLATION VERSUS MODELLING FOR CONTROL

P. M. Frank, E. Alcorta García and B. Köppen-Seliger
Dept. of Measurement and Control, Gerhard-Mercator-Universität Duisburg
Bismarckstr. 81 BB, D-47048 Duisburg, Germany
e-mail:p.m.frank@uni-duisburg.de

Abstract. The goal of this paper is to emphasize both the particularities of models needed for model-based fault detection and isolation (FDI) and the differences with respect to the models used in control. Of special interest is the question of complexity. This depends basically on the given situation such as the kind of plant, the kind and number of faults to be detected, the demands for fault isolation and robustness and the measurements available. However, in contrast to the wide-spread opinion that models for FDI have always to be more complex than those for control, the paper shows that diagnostic models for controllable and observable plants comprise only a partial description of the input/output model and are therefore less complex than those for control. This issue is discussed in terms of different model-based FDI approaches - analytical, data- and knowledge-based. As for the analytical approaches the necessary order of the diagnostic model is that of the transfer operator from the fault vector to the system output.

Introduction

A key issue in the design of fault tolerant control systems that aim at increased reliability and safety is *Fault Detection and Isolation (FDI)*, which has received much attention in the last two decades. A number of different FDI approaches making use of either output signal processing or of a model of the process have been proposed over the years. The most powerful approaches are those using a process model, where either quantitative, qualitative, knowledge-based or data-based models or combinations of them are applied.

The basic idea behind the model-based FDI approach is to take advantage of the nominal model of the system to generate residuals that contain information about the faults. Evidently, the quality of the model is of fundamental importance for both fault detectability and isolability and the avoidance of false alarms.

Since the early work of [3], system models common in control theory, which represent the dynamical behaviour between the system inputs and outputs (or states), have been taken for the design of FDI systems, although it is well known, that different kinds of applications require different types of models. For control system analysis and design, the system model has to represent the dynamical input-output behaviour of the system and should be as simple as possible. Hence, the model used is often drastically simplified and linearized ignoring many of the attributes of the physical nature of the system and only retaining the attributes that are deemed relevant for the behaviour of the resulting control system. Not so in FDI: here one needs a *representative* model of high fidelity and in some cases high preciseness which is in general of higher complexity than the one for control. But under certain circumstances, models for FDI can also be simpler than those for control, which has often been overseen in the FDI society.

The key point is that for FDI one needs only that part of the model which reflects the faults of interest and, with respect to robustness, is not or only weakly affected by disturbances and modelling uncertainty. Clearly, the resulting submodel highly depends upon several factors such as the kind and number of faults to be detected, the disturbances, uncertain parameters, and available measurements. Hence the first step of determining the reduced model for FDI is to find out that input/output measurable part of the system which contains the sources of the faults. Then, in a second step, this model can be reduced by confining on that part of it which is most strongly affected by the faults and most weakly influenced by the modelling uncertainties and unmodelled disturbances.

In this paper we focus our attention on the second step. For the different types of quantitative, qualitative and data-based models it is shown that for systems of minimal order (controllable and observable) and a specific set of faults to be monitored the required diagnostic model comprises in general only a

partial description of the original system. This issue is discussed in terms of different model-based FDI approaches. Note that the principal basic ideas of this paper are on the line with those discussed in [4], where the modelling for FDI in the context of fault isolation is the main concern. In addition to the ideas related to fault isolability, this paper focuses on the complexity of the models.

Features of Models for Residual Generation

The topic addressed in this section is the analysis of the features of the *representative* models for FDI compared to the models of control. Of special concern is the question of complexity. To the best knowledge of the authors, this issue has not been studied systematically so far. It has been considered previously only indirectly as a by-product of the results of specific approaches. Some general considerations are found in [4] and, for specific approaches some examples are given in [13] for the observer-based approach and in [6] for the parity space approach.

Our aim is to show from a more general point of view in what sense representative models for FDI are not coincident with the input/output models for control. First, it is evidenced that the diagnostic model can be of higher order than the input/output model for control if the system is not of minimal order (not controllable). Then it is demonstrated that in the case of minimal order systems the different model-based FDI approaches lead in general to partial descriptions of the system, i.e. to reductions of the input/output models. This will be discussed in terms of different types of modelling: analytical, data-based and knowledge-based.

Controllability versus Fault Detectability

As pointed out by [4], the fault isolation task can only be realized if the fault to be isolated has been previously taken into account in the model. In particular situations, the model used for control may not have all the information required for FDI. This is the case if the system is not controllable and the fault occurs in the uncontrollable part. Then the model for FDI must evidently be of higher order than the one for control in order to detect the faults.

To give an example, consider the following scenario: A system model which is not completely controllable w.r.t. the system input but observable has to be controlled and monitored. For control purposes a minimal realization based on Kalman canonical decomposition should be used. But then the uncontrollable modes of the original model are no longer present in the reduced model. The lost information (uncontrollable modes), however, is needed for FDI if a fault occurs in the eliminated (uncontrollable) states, see figure 1.

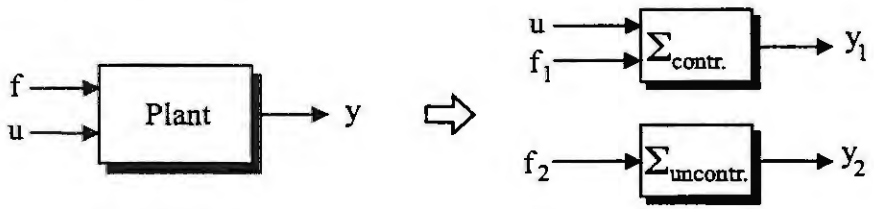


Figure 1: Controllable and uncontrollable subsystems under the influence of faults

Remark. The ideas of this subsection are not new. Under the name *Natural Redundancy* a residual generation approach taking advantage of the above ideas has been proposed for linear systems in [18]. A generalisation for nonlinear systems was considered in [25].

Analytical approaches

The approaches considered in this subsection make use of analytical (quantitative) models of the system. It will be shown that a reduced model for FDI compared to that for control is needed when we wish to detect only a limited set of faults, obtain robustness with respect to unknown inputs (disturbances, modelling uncertainty) and aim at selective detectability for the purpose of fault isolation. Then we need structured residual vectors, where each component should reflect only a subset of the fault vector; for this purpose only a part of the overall model of the process is needed.

Observer-based approach There is a number of different approaches for the design of diagnostic observers. These are the geometric methods [17], spectral theory [22], frequency domain [8] and algebraic methods [7, 13, 23]. In this contribution we follow the approach of [13] (see also [1]).

The basic idea is to find a state transformation of the given system such that the state can be divided into two parts: one independent of unknown inputs and the second one which is affected by the unknown inputs. Under the assumption that the set of states affected by the unknown inputs is obtainable from the output measurement of the system, the subsystem defined by the states which are not affected is obtained. Faults in this subsystem can be detected (isolated) by the use of an observer-based residual generator. The design is simple, because a Luenberger-like observer can be used.

The subsystem can be constructed as follows. First the vector f_k is partitioned into $\bar{f}_k$ and $\tilde{f}_k$. The vector $\bar{f}_k$ contains the set of faults whose effect on the subsystem is undesired and the vector $\tilde{f}_k$ contains the set of faults that will be monitored from this subsystem. Without loss of generality and to keep the analysis simple, the matrix D is neglected and no sensor faults are considered. The matrix E_a is partitioned into $E_a \triangleq \left[\begin{array}{cc} \bar{E} & E \end{array} \right]$ and the fault vector into $f_k \triangleq \left[\begin{array}{c} \bar{f}_k \\ \tilde{f}_k \end{array} \right]$.

Considering a non singular transformation matrix T such that $T\bar{E} = \left[\begin{array}{c} \bar{E}_1 \\ 0 \end{array} \right]$ and applying it as a state transformation $z_k = Tx_k$ to the system

$$x_{k+1} = Ax_k + Bu_k + E_a f_k; \quad y_k = Cx_k + Du_k + E_a f_k, \tag{1}$$

we have

$$z_{k+1} = TAT^{-1}z_k + TBu_k + T\bar{E}\bar{f}_k + TE\tilde{f}_k; \quad y_k = CT^{-1}z_k \tag{2}$$

With the partitions

$$z_k = Tx_k = \left[\begin{array}{c} z_{1k} \\ z_{2k} \end{array} \right]; \; TAT^{-1} = \left[\begin{array}{cc} A_{11} & A_{12} \\ A_{21} & A_{22} \end{array} \right]; \; TB = \left[\begin{array}{c} B_1 \\ B_2 \end{array} \right]; \; CT^{-1} = \left[\begin{array}{cc} C_1 & C_2 \end{array} \right]; \; TE = \left[\begin{array}{c} E_1 \\ E_2 \end{array} \right]$$

the transformed system can be written as

$$z_{1k+1} = A_{11}z_{1k} + A_{12}z_{2k} + B_1u_k + \bar{E}_1\bar{f}_k + E_1\tilde{f}_k \tag{3}$$

$$z_{2k+1} = A_{21}z_{1k} + A_{22}z_{2k} + B_2u_k + E_2\tilde{f}_k \tag{4}$$

$$y_k = C_1z_{1k} + C_2z_{2k} \tag{5}$$

Note that if $\text{rank}(C_1) = m - 1$ the disturbed state x_{1k} is eliminated from (4) and with this the desired subsystem can be obtained. Otherwise consider the singular value decomposition of C_1

$$C_1 = USV^T; \quad S = \left[\begin{array}{cc} \Sigma_1 & 0 \\ 0 & 0 \end{array} \right] \tag{6}$$

Define an output transformation T_1 as $T_1 = U^T$; Applying it to y_k results in

$$y_k^* = T_1y_k = \left[\begin{array}{cc} \Sigma_1 & 0 \\ 0 & 0 \end{array} \right] V^Tz_{1k} + T_1C_2z_{2k} \tag{7}$$

In order to eliminate a part of the unknown state z_{1k} in equation (4), consider $y_k^* = \left[\begin{array}{c} y_{1k} \\ y_{2k} \end{array} \right]$; $V^TC_2 = \left[\begin{array}{c} C_{21} \\ C_{22} \end{array} \right]$; $V^Tz_{1k} = \left[\begin{array}{c} z_{11k} \\ z_{12k} \end{array} \right]$, and z_{11k} can be obtained from the equation of y_{1k} as $z_{11k} = \Sigma^{-1}(y_{1k} - C_{21}z_{2k})$. Substituting it into (4) results in

$$\begin{aligned} z_{2k+1} &= A_{211}\Sigma^{-1}y_{1k} + (A_{22} - A_{211}\Sigma^{-1}C_{21})z_{2k} + B_2u_k + A_{212}z_{12k} \\ y_{2k} &= C_{22}z_{2k} \end{aligned} \tag{8}$$

where $A_{21}V \triangleq \left[\begin{array}{cc} A_{211} & A_{212} \end{array} \right]$. In order to have (8) insensitive to $\bar{f}_k$, the elimination of z_{12k} from (8) is required. To do so, the above procedure has to be applied to (8) once more assigning $z_{12k} \rightarrow \bar{f}_k$. The procedure is continued until the undesired faults have lost their effect on the calculated subsystem. The procedure is illustrated in figure 2.

After the subsystem sensitive to the desired group of faults has been found, a residual generator is obtained by designing an output observer for the subsystem (8). The residual is defined as the difference between the output of the system and the estimated output, see figure 3.

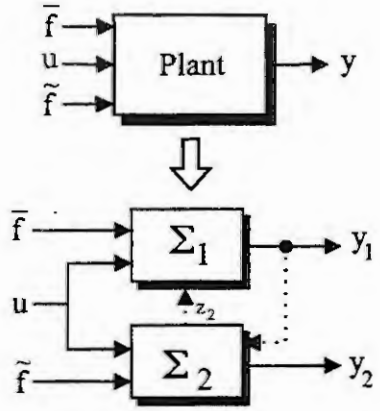


Figure 2: Decoupling of unknown inputs in observer-based residual generation

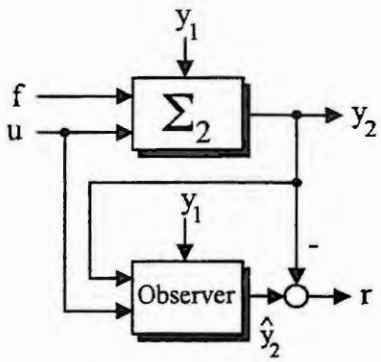


Figure 3: Observer-based residual generator for decoupled subsystem

Parity Space

The residual in the parity space approach is either defined based on the input output operator representation of the system (1) as

$$r_k = W(z)(y_k - G_u(z)u_k) = W(z)G_f(z)f_k \tag{9}$$

with $W(z)$ representing a filter yet free to select, or based on the state space representation as

$$r_k = \underbrace{v_1 z^{-s}|zI - A|}_{\triangleq W(z)} . G_f(z)f_k \tag{10}$$

The transformation matrix v_1 can be used to reduce the set of the parity equations to those that only contain information about the faults and/or are only little or not affected by the uncertain parameters.

As can be seen from (9), (10), the residual in the parity space can be expressed as a relationship of inputs, nominal (fault free) model and outputs. The supervision of faults can be carried out in a direct way by selecting an adequate parity matrix (vector). If multiple faults have to be detected and isolated, the use of structured residuals is mandatory. Structured residuals for parity space have for example been studied by [9] for the I-O approach and, for example, by [24] for the state space approach.

The basic idea is to design a set of residuals, each of them sensitive to a different set of faults. As can be seen from (9) and (10), this can be achieved by a proper choice of $W(z)$ or v_1, respectively. Hence, for the generation of the residual r_k we need only a model of the form $W(z)G_u$ or $v_1 z^s \mid zI - a \mid G_u$, resp., instead of G_u. This leads to a model reduction of the original nominal model G_u.

An example taken from [6] may illustrate this result. Consider the system

$$\begin{aligned} x(k+1) &= \begin{bmatrix} a_{11} & a_{12} \\ 0 & a_{22} \end{bmatrix} x(k) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(k) \\ y(k) &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x(k) \end{aligned} \tag{11}$$

and suppose that both the actuator and the component faults have to be supervised. Following [6], there are three linearly independent parity equations

$$y_1(k) - (a_{11} + a_{22})y_1(k-1) + a_{11}a_{22}y_1(k-2) - a_{12}u(k-2) = 0 \tag{12}$$

$$y_1(k) - a_{11}y_1(k-1) - a_{12}y_2(k-1) = 0 \tag{13}$$

$$y_2(k) - a_{22}y_2(k-1) - u(k-1) = 0 \tag{14}$$

Note that for the supervision of a_{22} and of the actuator only the parity equation (14) is required, which represents only a partial description of the original system (11). In a similar way, for the supervision of

a_{11} and a_{12} the parity equation (13) can be used, which is another partial description of (11). To select the proper equations means to find v_1 in (10), for which task a number of methods have been developed, see, e.g., [6].

Remark. An easy and systematic way to design structured residuals is by applying the same procedure used to get a robust subsystem in the observer-based approach and building a parity space model for the resulting subsystem.

Parameter Estimation

Parameter estimation (PE) as developed for and widely used in modelling, signal processing and control can also be applied to FDI [14]. For control applications, both the model structure and the parameters of the plant are identified either off-line (for controller design) or on-line (for controller adaptation). The original possibly complex nonlinear physical system is usually approximated by linear differential or difference equations valid in a certain range of operation. The physical meaning and the internal relationships of the original system parameters are not important, i.e., the mathematical parameters are sufficient for this task.

The basic idea behind the application of parameter estimation to FDI is the on-line estimation of the parameters of the actual system model and the comparison to their nominal values. The resulting deviations are the residuals used for FDI [14]. For the interpretation of the faults we need in general the deviations in terms of the physical parameters. This is why the model should be as detailed as possible. In other words, it is not preferable to simplify the nonlinear functions into linear ones etc..

Most systems in engineering are continuous in time. The mathematical parameters are here closer related to the physical parameters than in a discrete time model. Hence the parameter estimation for continuous time models bears special findings for FDI. The linearization of nonlinear functions and the discretization of the differential equations make the relationship between mathematical parameters and physical parameters more complicated.

It can be concluded that in this case the model for FDI must be more complex than that for control system design. In return, these arduous efforts bring along the following advantages:

- This approach can provide a deeper insight in the system. With a bit more efforts (e.g. adaptive control), the effects of the faults can be compensated in order to gain fault-tolerance.
- When the relationship between the model parameters and physical parameters is unique, the fault isolation is easy to implement.
- This method also provides direct fault identification because the physical parameter deviations stand directly for the severity of the faults.

The main limitation of this approach is that the estimated parameters must be persistently excited by an input signal and that for the isolation of the faults the relationship between the physical and mathematical parameters must be unique. It is, however, also possible to utilize directly the mathematical parameters (equation coefficients or zeros and poles of the transfer functions) [10, 11]. In this case, simplified models can be used, because only those parameters which are related to the possible defect parts in the system need to be estimated, whereas the other parameters can be taken as constant. This means that only a partial model is required. In practice, the original system is divided into subsystems as small as possible, and with only few parameters to be estimated. An extra bonus of the reduction and partition of the original model is that the requirements for the excitation signals can be relaxed. This also speeds up the estimation convergence.

Data-based approaches

Data-based models constitute an alternative to the analytical approach to FDI when the latter is either not available or not feasible. In many practical applications large archives of process data exist which can be used to set up a data-based model. The given set of input-output data of the possibly nonlinear system can be used to train a properly prestructured nonlinear model. The learning can be achieved by adaptation due to a given performance index.

Data-based models designed for FDI principally aim at the (best possible) estimation of those output measurements which are influenced by the faults of interest. This means that the resulting models have

a different input and output measurement space compared to the *functional* models for control. This can lead to less complexity. However, the data-based models are usually nonlinear in contrast to the functional analytical models which are often linear and hence less complex. Let us consider the two most common types of data-based models are neural networks and fuzzy relational models.

The neural network approach

Basically, the artificial neural net (ANN) represents a nonlinear system refereing to its input-output behaviour. The nonlinear transformation results from its inner structure. In general the ANN consists of neurons, simple processing elements, which are activated as soon as their inputs exceed a certain threshold. The neurons are arranged in layers which are connected such that the signals at the input are propagated through the network to the output. The choice of the transfer er function of each neuron (e.g. sigmoidal function) yields the nonlinear overall behaviour of the network. During a training period a set of parameters of the neural network is learned from a given set of data aiming at the "best" approximation of the behaviour of the system. Since the late eighties artificial neural networks have been studied for FDI. First only slowly varying processes ([12], [20], [21]) were considered, but due to recent efforts to model nonlinear dynamic systems [5], FDI can greatly benefit from this. The training is generally performed using measurements from the fault free process.

For residual generation purposes the neural network simply replaces the analytical model describing the process under normal operation [16]. Employing a nonlinear input/output description

$$\underline{y}(k) = g(\underline{y}(k-1),...,\underline{y}(k-q),\underline{u}(k),...,\underline{u}(k-p)) \tag{15}$$

the neural network approximates the nonlinear vector function $g(\cdot)$.

This general pattern may be substantially compressed considering that

- it is useful to estimate each system output with a seperate neural network
- not all of the system outputs may be influenced by the faults under consideration and need to be estimated.

The first point implies that for each output the neural network's input space is chosen differently, each leading to the best possible approximation of the respective output. This choice includes the different inputs and outputs from the available ones and the number of tapped delays for each signal (figure 4).

As an example the estimates for a system with two inputs and three outputs can look as follows:

$$\begin{aligned}
\hat{y}_1(k) &= g_1(y_1(k-1), y_2(k-1), u_1(k)) \\
\hat{y}_2(k) &= g_2(y_2(k-1), y_2(k-2), y_3(k-1), u_2(k)) \\
\hat{y}_3(k) &= g_3(y_1(k-1), y_2(k-1), y_3(k-1))
\end{aligned}$$

The appropriate choice of the input space is one of the most difficult tasks when configuring the neural network. One either needs to have enough process knowledge and then use a trial and error strategy or has to apply an optimization algorithm such as a genetic algorithm [16].

Keeping in mind that for FDI only those outputs need to be estimated which are affected by faults one can conclude that neural models for FDI may be of reduced order and hence of less complexity due to an adequate input space in comparison to functional models used for control.

Residual generation based on fuzzy relational models

The approach considered here has been described in [2], [19]. The fuzzy observer is founded on a fuzzy relational model of the process, which is formed by the composition operator (T-conorm/T-norm operator) applied to a fuzzy relational matrix R defining the relation between process input and output, and the fuzzy cartesian product of the fuzzified input-output signals and its delays on a time window.

A mathematical representation of the fuzzy observer is given by

$$\hat{Y}_i = \mathrm{R}_0 \circ \mathrm{X} \tag{16}$$

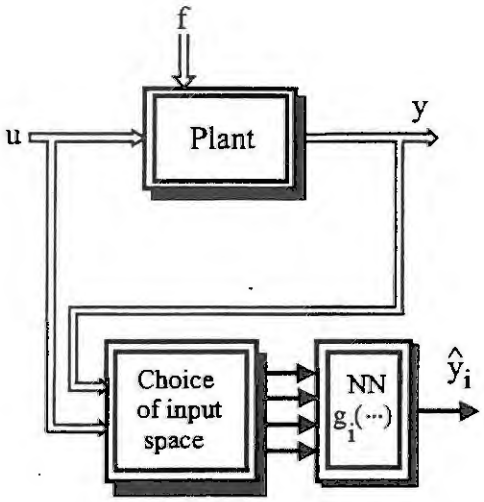


Figure 4: Separate estimation of output signals by ANN

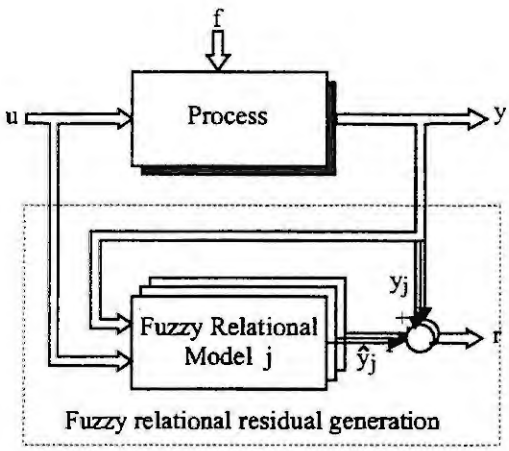


Figure 5: Fuzzy observer-based residual generation

where $\hat{Y}_i$ and X are given in the fuzzy space. X is defined by the fuzzy cartesian product of the input-output measurements and its delays, i.e.

$$\mathrm{X} \;=\; \mathrm{U(k)} \times \cdots \times \mathrm{U(k\text{-}n)} \times \mathrm{Y}_i(k) \times \cdots \times \mathrm{Y}_i(k-n) \tag{17}$$

and R_0 is the relational matrix of the nominal (fault free) model. The fuzzy residual generator as shown in figure 5 determines the difference between the measured and the estimated output using a fuzzy output observer.

In order to show that models for FDI used in the fuzzy output observer can be different from functional ones, we consider a (nonlinear) system, whose inputs and outputs are represented by u_k and y_k, respectively. Suppose we are interested to detect a fault in the j^{th} output sensor. For this we need a residual generator based on fuzzy relational models which is sensitive to faults in the j^{th} sensor. This can be constructed as follows.

Consider the vector X,

$$X \;=\; U_k \times \cdots \times U_{k-\mu_1} \times Y_{jk} \times \cdots \times Y_{jk-\mu_2}, \tag{18}$$

which represents a combination of all premise variables appearing in the premises of the rules that are represented by the relational matrix R. The variables U_k and Y_k are the fuzzified values of u_k and y_k, respectively.

A relational fuzzy model for the j^{th} output is given by

$$Y_{jk} \;=\; R_0 \circ X \tag{19}$$

To obtain the relational matrix R_0, a fuzzy relational equation has to be solved. However, in general, the set of solutions is empty [2]. The alternative approach is to find a solution R_a such that the output of the relational model (19) when R_a is used is an approximation $\hat{Y}_{jk}$ of Y_{jk} in the sense of a given criterion $J(R_a)$, e.g., the quadratic error in the real valued space

$$J(R_a) \;=\; \frac{1}{2}\sum_{k=1}^{N}(y_{jk} - \hat{y}_{jk})^2, \tag{20}$$

where N is the number of elements in the learning set and the quantities used in the criterion are the defuzzified corresponding values of the outputs.

The above procedure shows that the representative model used for the design of a residual generator based on relational models is a partial description of the original system.

Note that the structure of the residual generator based on fuzzy relational models is similar to the one based on neural nets. A basic difference is that the signals used for the fuzzy relational model have been previously fuzzified, for the neural net approach the measured signals are utilized directly.

Knowledge-based approaches

Another alternative to the analytical FDI approach is the knowledge-based approach which makes use of the knowledge available to derive either a qualitative description of the system in the form of a qualitative model or a rule-based representation.

In this paper, we restrict ourselves to qualitative models on the basis of qualitative differential equations following [26]. Other approaches, as for example the important concept of fuzzy rule-based modelling, are not considered here.

Before discussing details of the qualitative models some general remarks on qualitative modelling strategies should be made. In order to perform totally reliable fault diagnosis and to avoid false alarms an *accurate* model is of absolute importance. However, an accurate model does neither mean the highest precision in the description nor the highest complexity of the model [15]. Developing a model on a higher level of abstraction can still lead to an accurate model though less complex. On the other hand *preciseness* of measurements is only of critical importance for diagnostic tasks if the models are very precise in their description as well. An increased imprecision of the measurements can be tolerated by models on a higher level of abstraction still leading to a correct fault decision. In exchange with this advantage in modelling effort one has to put up with the drawback of less sensitivity to smaller faults.

Qualitative residual generation

When complete information about an industrial process is not available, the quantitative model-based techniques for FDI can be replaced by qualitative ones that make use of the available incomplete information by building a qualitative model, in terms of which the analysis and reasoning can be carried out. Different from the other model-based approaches, a qualitative model may be based on *qualitative differential equations* (QDE). A QDE for a specific system has the same structure as the corresponding ordinary differential equation that models the dynamic system in continuous time. However, the information about the parameters is only of "semi-quantitative" nature being frequently only partially known or uncertain.

As a result, a constrained model is obtained which consists of qualitative variables representing the physical parameters of the system and a set of constraints of how those parameters are related to each other. The behaviour can be described by a graph consisting of the possible future states of the system. Due to the inherent ambiguity of the qualitative representation and calculus, the simulated possible behaviour is in general not unique, but could take any path through the graph starting at the initial state, as shown in figure 6.

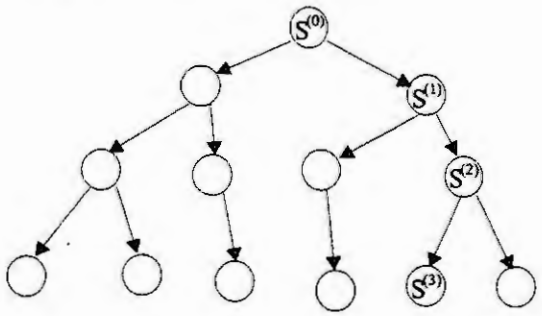


Figure 6: Graphical description of behaviours

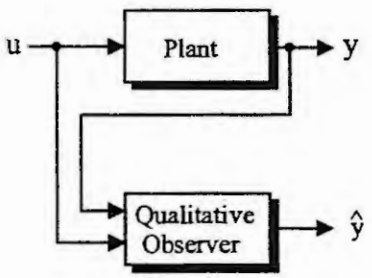


Figure 7: Qualitative output observer

A qualitative observer (QOB) used for residual generation makes use of qualitative simulation on the basis of conventional filtering techniques to perform an *observation filtering* [26]. The principle of *observation filtering* is that the simulated qualitative behaviour of a variable must cover its counterpart of the measurement obtained from the system itself, otherwise the simulated behavioural path is inconsistent and can be eliminated [26]. A scheme of the qualitative observer is shown in figure 7.

The idea behind the qualitative observer-based FDI is that a fault causes a deviation of the system output in such a way that its counterpart of the estimated output is no more consistent, i.e. a fault will produce an empty set of qualitative estimated states, which is impossible in a fault free case. A model for FDI can now be derived following the basic idea of using fuzzy relational models. To show

this, consider a nonlinear system described by the QDE

$$[\dot{x}] = f([x],[u]); \quad [y] = C[x] \tag{21}$$

where $[x] \in \Re^n$ is the system state, $[u] \in \Re^p$ is the system input and $[y] \in \Re^m$ is the system output. If we are only interested in the detection of a single sensor fault, i.e. in the i^{th} sensor, a qualitative observer can be designed obeying the equation

$$[\dot{x}] = f([x],[u]); \quad [y_i] = C_i[x] \tag{22}$$

where $[y_i]$ is the i^{th} element of $[y]$ and C_i is the i^{th} row of C. A more detailed description of the design of qualitative observers can be found in [26].

Conclusions

The most common model-based FDI approaches have been examined for their modelling efforts, accentuating the differences between models used for FDI and those used for control. Firstly, we have studied the role of the transfer operator from the fault vector to the system output and its decisive impact on the residual generation with respect to fault isolation. The similarities of the different analytical FDI approaches have been underlined, and the dependence of the residual upon the transfer operator from the fault to the output has been discussed in terms of complexity. As a main result it has been shown that for robust FDI of controllable systems and making use of quantitative, data-based or qualitative models, the model needed comprises only a partial description of the system and is thus not simply identical with the model for control. Mostly it is less complicated. Even though the most common model-based approaches have been taken into account in this paper, we do not claim completeness of our study. Some important concepts such as the continuous-time parity space approach, the continuous-time dead-beat observer approach, analytical nonlinear FDI methods and stochastic FDI approaches have not been included into this consideration. However, it seems to be evident that they do not require fundamental revision of the message given in this paper.

Ackowledgement. The authors want to thank the Deutscher Akademischer Austauschdienst (DAAD) for the financial support of this work. Moreover, the authors are grateful to Z. Han, Z. Zhuang and P. Amann for helpful discussions and support in the preparation of the paper.

References

[1] E. Alcorta-García and P. M. Frank. Analysis of a class of dedicated observer schemes to sensor fault isolation. *UKACC International conference on Control'96*, pages 59–64, Exeter, UK 1996.

[2] P. Amann and P. M. Frank. On fuzzy model-building in observers for fault diagnosis. In *Proceedings of the 15th World Congress on Scientific Computations, Modelling and applied Mathematics IMACS '97, Berlin.*, August 24-29 1997.

[3] R. V. Beard. *Failure Accomodation in Linear Systems Through Self-Reorganization.* Ph.D. Dissertation, M.I.T., 1971.

[4] M. Blanke. Consistent design of dependable control systems. *Control Eng. Practice*, 4(9):1305–1312, Sep. 1996.

[5] S. Chen, S. A. Billings, and P. M. Grant. Recursive hybrid algorithm for non-linear system identification using radial basis function networks. *International Journal of Control*, 55(51):1051–1070, 1992.

[6] E. Y. Chow and A. S. Willsky. Analytical redundancy and the design of robust failure detection systems. *IEEE Trans. on Autom. Control*, AC-29(7):603–614, July 1984.

[7] P. M. Frank. Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy - a survey. *Automatica*, 26:459–474, 1990.

[8] P. M. Frank and X. Ding. Frequency domain approach to optimally robust residual generation and evaluation for model based fault diagnosis. *Automatica*, 30(5):789–804, 1994.

[9] J. Gertler. Structured residuals for fault isolation, disturbance decoupling and modelling error robustness. In *IFAC-On line fault detection and supervision in the chemical process industries, Newark, Delaware, USA*, pages 111–119, April 1992.

[10] J. B. Gomm, D. Williams, and P. Harris. Detection of incipient faults using approximate parametric models. In *IFAC-On line fault detection and supervision in the chemical process industries, Newark, Delaware, USA*, pages 138–143, April 1992.

[11] Z. Han and P. M. Frank. Fault detection via characteristic parameter estimation. In *Proc. of the* 3^{rd} *European Control Conference*, pages 378–382, Italy 1995.

[12] D. M. Himmelblau. Use of artificial neural networks to monitor faults and for troubleshooting in the process industries. In *IFAC Symposium On-line fault detection and supervision in the chemical process industry, Newark, Delaware, USA*, 1992.

[13] M. Hou and P. C. Müller. Fault detection and isolation observers. *Int. J. of Control*, 60:827–846, 1994.

[14] R. Isermann. Process fault detection based on modeling and estimation methods-A survey. *Automatica*, 20:387–404, 1984.

[15] Uwe E. Keller. *Qualitative Model refernece Adaptive Control.* PhD thesis, Heriot-Watt University, Department of Computing & Electrical Engineering, 1999.

[16] B. Köppen-Seliger and P. M. Frank. Fault detection and isolation in technical processes with neural networks. In *Proc. 34th Conf. on Decision and Control CDC'95, New Orleans*, 1995.

[17] M. Massoumnia. A geometric approach to the synthesis of failure detection filters. *IEEE Trans. on Autom. Control*, AC-31(9):839–846, Set. 1986.

[18] L. A. Mironovskii. Functional diagnosis of dynamic systems. *Automation and Remote Control*, pages 96–121, 1980.

[19] R. Querelle, N. Mary, N. Kiupel., and P. M. Frank. Use of qualitative modelling and fuzzy clustering for fault diagnosis. *WAC'96*, pages 527–532, Montpellier, France. May 1996.

[20] T. Sorsa, Suontausta, and H. N. Koivo. Dynamic fault diagnosis using radial basis function networks. In *TOOLDIAG'93, Toulouse*, April 5-7 1993.

[21] S. G. Tzafestas and P. J. Dalianis. Fault diagnosis in complex systems using artificial neural networks. In 3^{rd} *IEEE Conference on Control Applications, Glasgow*, pages 877–882, August 24-26 1994.

[22] J. E. White and J. L. Speyer. Detecton filter design: Spectral theory and algorithms. *IEEE Trans. Automat. Control*, 32(7):593–603, July 1987.

[23] J. Wünnenberg. *Observer-Based Fault Detection in Dynamic Systems.* VDI-Fortschrittsber., VDI-Verlag, Reihe 8, Nr. 222, Düsseldorf, Germany, 1990.

[24] D. Yu and D. N. Shields. A fault isolation method based on parity equations with application to a lathe-spindle system. *UKACC International conference on Control'96*, pages 317–322, Exeter, UK 1996.

[25] A. N. Zhirabok. Natural redundancy and fault detection and isolation in dynamic systems. In *Proc. IFAC Symp. SAFEPROCESS '94, Espoo Finland*, pages 335–340, June 1994.

[26] Z. Zhuang and P. M. Frank. Qualitative observer and its application to FDI systems. *Proc. Instn. Mech. Engrs.*, 211(4):253–262, 1997.

AUDITORY DISPLAYS IN HUMAN-MACHINE INTERFACES OF MOBILE ROBOTS FOR NON-SPEECH COMMUNICATION WITH HUMANS

Gunnar Johannsen
IMAT-Lab. Systems Engineering and Human-Machine Systems
University of Kassel, Moenchebergstr. 7, D-34109 Kassel, Germany
joh@imat.maschinenbau.uni-kassel.de

Abstract. Auditory displays are developed and investigated for mobile service robots in a human-machine environment. The service robot domain was chosen as an example for future use of auditory displays within multimedia process supervision and control applications in industrial, transportation, and medical systems. The design of directional sounds and additional sounds for robot states as well as the design of more complicated robot sound tracks are explained. Basic musical elements and robot movement sounds are combined. An experimental study on the auditory perception of sound tracks for the predictive display of intended robot trajectories in a simulated supermarket scenario is described.

Introduction

The movements of robots produce natural sounds. These can communicate information about the actual states and trajectories of the robot to the human. Such auditory information is particularly useful with mobile robots where the sounds indicate the actual positions and movements in space. This information is often also important in other technological processes. However, the auditory information is normally used by the human only in addition to the visual information. Thus, the question arises in which way can auditory information be applied even beyond the natural sounds in future multimedia human-machine interfaces.

Multimedia technologies are currently available which can more systematically be applied for human-systems communication in industrial, transportation, medical, and many service domains [1]. Human-machine interfaces have generally reached a very mature development status [3], [5]. However, their presentational level is often restricted to the visual channel. Taking the multimedia concept more seriously allows to re-integrate the most important human sensory modalities for the supervision and control of technical processes and systems. Thus, many drawbacks and restrictions of over-emphasizing the visual channel can be corrected by the additional appropriate use of auditory information.

However, before developing the next generation of multimedia human-machine interfaces for process supervision and control applications, more knowledge on suitable auditory displays needs to be collected. Even the investigation of auditory warning displays needs to be further intensified [7]. More ambitious auditory displays will go beyond the warning displays by communicating systems states and intentions by means of semantic sound symbols and sound tracks.

This paper reports on results from a recent research project[1] in which auditory displays for autonomous mobile service robots in a human-machine multi-agent environment have been investigated and developed. The domain of mobile service robots has been chosen as an example application [6]. The results will contribute to the future use of auditory displays within all kinds of multimedia process supervision and control applications. The idea of the auditory displays for the service robots is to combine relevant noise signals of their movements with basic musical elements to intelligible auditory symbols. Co-operative mobile robots shall communicate their actual positions, movements and intentions as well as special states by means of non-speech audio symbolic expressions to the human.

In the next two sections, the design of directional sounds for robot movements and special robot state sounds as well as complete robot sound tracks will be explained. The auditory perception of intended robot trajectories in a supermarket scenario will subsequently be described.

---

[1] The research project on "The Importance of Acoustical Information for the Guidance and Usage of Technical Systems" was supported by the VW-Stiftung (Volkswagen-Foundation) and the University of Kassel during my sabbatical research semester. It was carried out in Vienna, Austria from March to October 1999. My main host was the Institute for Handling Devices and Robotics at the TU Wien (Vienna University of Technology – Univ.-Professor Dr. P. Kopacek). Further, a co-operation existed with the Institute of Electro-Acoustics, Experimental and Applied Music at the University of Music and Performing Arts, Vienna.

Design of auditory displays for directions of motion and robot states

The directions of motion of the robot in space can be considered with a more fine-grained resolution of 48 directions around the circuit of a compass card of 360°. So far, the subset of eight directions of motion with corresponding newly designed directional sounds has been tested.

These eight directions are the four main directions of Left, Up, Right, and Down as well as the intermediate directions of Down-Left, Up-Left, Up-Right, and Down-Right. Each directional sound consists of three tones. The musical basic elements rhythm and melody are used in the four main directions, independently of each other. The directional sound Up is represented by a melody upwards whereas a melody downwards denotes the sound Down. In both cases, each tone is of equal time duration. A rhythm of two short tones followed by one long one, all on the same pitch level, means Left. Consequently, a rhythm with one long tone followed by two short ones on the same pitch level expresses the direction Right. The musical elements melody and rhythm are combined in the intermediate directions with respective intermediate values of melody span and rhythm. Each of the eight directions is presented in four variations, with changed sound colour (timbre) or changed tempo. Music instruments and robot noises are used. The understandability and the recallability of these directional sounds has been tested with non-musicians and with musicians [4].

All sounds were created with a powerful PC and Windows 95, Logic Audio software, the Cool Edit audio editor, a MIDI synthesizer, and a keyboard. The pure musical sounds from the synthesizer have been recorded with the Yamaha DSP (Digital Signal Processing) Factory with its audio expansion unit under the Logic Audio software on the PC. Thus, wav-files have been produced which needed some minor audio editing.

The equivalent sounds of robot noises have been derived from DAT recordings of the movement noises of real robots in the laboratory. Many different original sounds from robot movements, navigation, and related activities have been recorded from two mobile service robots.[2] The variation of the directional sounds which is based on robot noises has been generated through several steps of audio editing from the recordings of the movements of one of these service robots. The same melody and rhythm patterns are composed by time and frequency editing as in the pure musical sound cases. The Logic Audio editor has been used for frequency transpositions of the pitch levels as requested in the different tones of the directional sounds. The Cool Edit audio editor was more appropriate for cutting and assembling the necessary time slices of each sound element for the desired rhythms of the directional sounds. Some amplifications with fading-in and fading-out effects have been performed for achieving a clear separation between the different three tones of each sound.

Additional sounds for robot states and situations have also newly been designed. These robot states and situations are Heavy Load, Waiting, Near Obstacle, and Low Battery. The latter sound was recorded from the original robot's indication of the low battery status. This is a continuous, quite annoying high tone. The other three sounds have been played on the MIDI keyboard. The author tried to convey the subjective impression of the meanings of these three sounds. For example, the Heavy Load sound was played with three parallel tubas as one accentuated short time-interval tone followed by one tone of a longer time duration.

Design of robot sound tracks

A simulated supermarket scenario was designed in such a way that a mobile service robot can make straight movements and turnings of 45 and 90 degrees. Sound tracks for the predictive display of intended robot trajectories are composed of moving-straight and turning sounds. The moving-straight segments are represented by the directional sounds which have been described in the preceding section. The directions Left, Right, Up, and Down are particularly used but also a few of the intermediate directions (Down-Left, Up-Left, Up-Right, and Down-Right) are sometimes possible. Down means downwards on the computer screen and towards the human subject, shown on the lower middle of the screen. Correspondingly, Up means away from the subject.

The turnings are derived from recordings of the original robot turning sound by transposition. They are always heard with any directional change between any kind of two complete moving-straight sections. If a complete moving-straight section consists of a number of straight segments of the same direction, the appropriate directional sound is repeated correspondingly without any turning sound in between. A segment is defined as the straight connection between two neighbouring active decision areas; see next section.

The robot sound tracks actually used in the supermarket scenario are the overlappings of the sound tracks of the intended robot trajectories and, during some of their segments, the additional sounds for robot states and

[2] http://www.ihrt.tuwien.ac.at/IHRT/English/mob_rob.htm (Laboratory for Mobile Robots – Univ.-Professor Dr. P. Kopacek)

situations which have been described at the end of the preceding section. In some of the robot sound tracks the sounds of the real robot movements are also overlayed.

Intended and perceived robot trajectories

Intelligible auditory symbols and sound tracks are presented to human subjects in experiments with a supermarket scenario. The supermarket is realized with a simulated (Windows 95 and Delphi) environment of a mobile service robot [2]. It is assumed that the supermarket is open during seven days a week for 24-hours. A mobile service robot for cleaning and for carrying goods will inform the human subject (the customer) with sound symbols of non-speech auditory predictive displays about the trajectory of its intended movements and about the additional robot states and situations Heavy Load, Waiting, Near Obstacle, and Low Battery. These additional sounds for the robot situations have to be learned by the human subjects in a training phase at the beginning of the experiments. The subjects can listen to these sounds in any order as often as they wish.

A floor plan of the supermarket is visualized on the computer screen. The human subject is shown in turquoise on the lower middle and the robot in different starting positions which are depending on the investigated trajectory; see Figure 1.

A matrix of decision areas has been constructed. Any intersection between a horizontal and a vertical corridor together with the respective nearest surrounding of this crossing, in which alternative routes can be chosen (beyond returning the same way), is determined as an active decision area in the visual floor plan of the supermarket; see Figure 1.

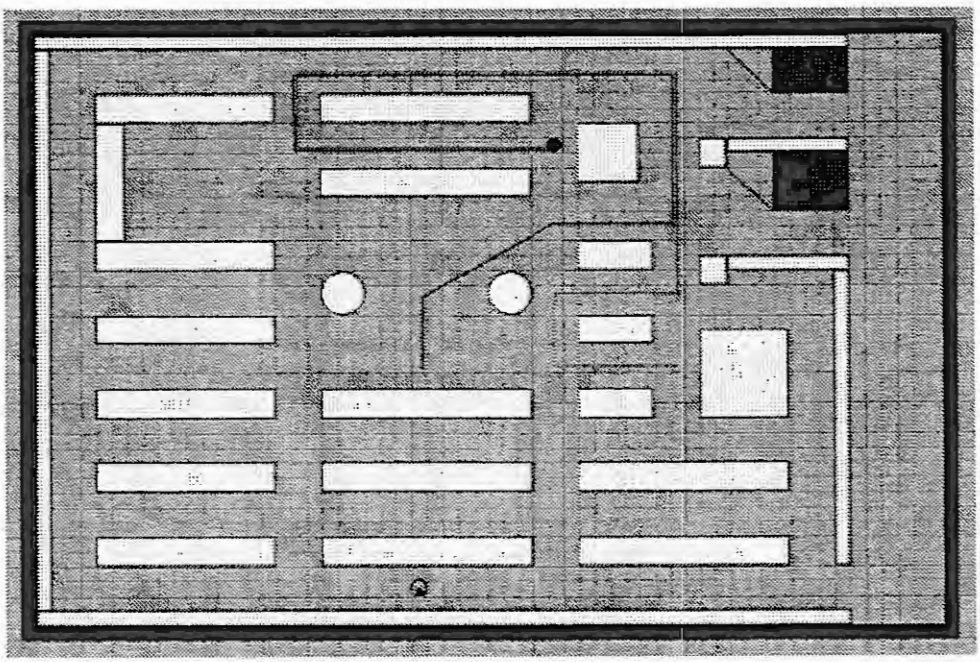


Figure 1. Replay of intended and perceived robot trajectories in the supermarket scenario.

The human subjects are asked to recognize and to understand the intended trajectory of the robot indicating the intended directions where the robot plans to move on, as well as the overlapped additional sounds of the robot situations, from listening to the robot sound track. They have to draw the auditorily perceived trajectory into the visual floor plan of the supermarket on the computer screen and have to mark the perceived additional sounds. The subjects have been informed about the correctness of their auditory perception with respect to the trajectories.

Altogether, the subjects perform four experiments, each with four different trajectories. The intended trajectory and the perceived trajectory as well as the intended and the perceived additional sounds are recorded.

Also, the durations of the training phase and of the drawing of each perceived trajectory are measured. The intended and the perceived trajectories can be compared in the replay mode; see Figure 1.

In the last two of these experiments, the sound tracks of the trajectories are composed of the same sound symbols of the intended trajectories and the overlayed additional sounds for the robot situations. However, the sounds of the real robot movements are now also overlayed. This makes the scenario even more difficult for the subjects but it is also more realistic. In a real-world human-robot environment, the real robot movements are also always heard. They are presented in real time whereas the overlayed intended trajectories are auditory predictor displays and, thus, faster than real time.

The experimental results with eight non-musicians and two professional musicians showed large differences in their auditory perception. Three of these ten subjects (the two musicians and one of the non-musicians) made only very few errors with the perception of the robot sound tracks of all trajectories as well as the additional sounds for the robot states and situations.

Conclusions

The design of directional sounds, robot state sounds, and robot sound tracks has been accomplished with basic musical elements and recorded robot noise signals. The experimental study showed that the sound symbols and sound tracks can well be perceived and are understandable, at least for more musical people. Positive training effects have been observed with all human subjects. The investigated sound tracks are feasible means of communication in human-machine interaction with mobile robots. Similar sound tracks can possibly be designed also for other application domains.

References

1. Borys, B.-B., and Johannsen, G., An experimental multi-media process control room. In: Proc. Annual Conference 1997: Human Factors and Ergonomics Society Europe Chapter, Advances in Multi-media and Simulation, Bochum, 1997, 276-289.
2. Fürst, M., Ein neues übergeordnetes Konzept zur integrierten Navigation mobiler Roboter. Diplomarbeit, Institut für Handhabungsgeräte und Robotertechnik, TU Wien, 1999.
3. Johannsen, G., Cooperative human-machine interfaces for plant-wide control and communication. In: Annual Reviews in Control, (Ed.: Gertler, J.J.). Pergamon, Elsevier Science, Oxford, 21 (1997), 159-170.
4. Johannsen, G., Analysis of audio symbols based on musical and robot-movement sounds using time-frequency methods. In: Proc. Diderot Forum on Mathematics and Music – Conference on Computational and Mathematical Methods in Music, University of Vienna, 1999.
5. Johannsen, G., Ali, S., and van Paassen, R., Intelligent human-machine systems. In: Methods and Applications of Intelligent Control, (Ed.: Tzafestas, S.G.) Kluwer, Dordrecht, 1997, 329-356.
6. Kronreif, G., Probst, R., and Kopacek, P., Modular service robots – State of the art and future trends. In: Proc. 8th International Conference on Advanced Robotics ICAR'97, Monterey, CA, 1997, 51-56.
7. Stanton, N.A. and Edworthy, J. (Eds.), Human Factors in Auditory Warnings. Ashgate Publishing, Aldershot, 1999.

STATE AND PERSPECTIVES OF USER INTERFACES IN AUTONOMOUS MOBILE ROBOTS

F. Matía and A. Jiménez
DISAM - Univ. Politécnica de Madrid
José Gutiérrez Abascal 2, E-28006 Madrid (SPAIN)
e-mail: matia@disam.upm.es

Abstract. The paper reviews the state of user interfaces in present mobile robotics applications. For sure, user interfaces include graphical environments (Man-Machine Interfaces) but along the paper we analyze them from a more general point of view, including also network communications, remote sensor management, high and low level robot command interfaces, etc. After the revision, we describe new perspectives that have arisen from the use of new specific hardware and software development languages and tools, giving some examples which go from the case of autonomous multirobot applications, to teleoperation applications.

1 Introduction

Along the paper we will consider Man-Machine Interfaces at the center of discussion. Nevertheless, and from our point of view, a user interface is a more general concept, since there exist different kind of robot users which expect different things from it: i) the programmer needs a nice set of tools to develop his/her control programs, ii) the application user needs to send a set of tasks to be done (perhaps by a set of mobile robots), and iii) the robot operator needs a powerful Man-Machine Interface (MMI) to send commands as well as to visualise the information on the screen. Figure 1 shows a typical operation scheme of a mobile robot.

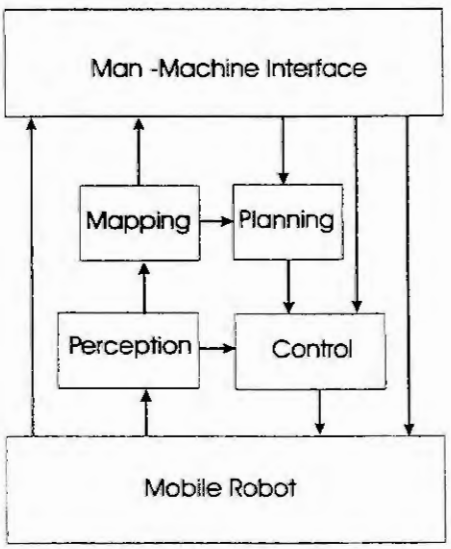


Figure 1: Robot operation scheme

At the top, we have the user itself, while at the bottom we have the mobile robot. In between them we have the navigation modules (i.e. software) that controls the robot movements and decisions:

- the perception module is in charge of processing raw sensor data and transforming them into: an accurate position estimation, some acurate world maps, and filtered proximity data for the obstacle avoidance system. Some of this information may be visualised in the MMI.
- the mapping module is able to manage geometric and occ-grid maps as well as quat-trees, which can also be visualised by the user.

- On the right side the planning module generates a priori free obstacle trajectories wich will be the reference for the control module.
- The control module must follow the previous trajectory, avoiding obstacles using the information supplied by the perception module, and generation velocity commands to the robot.

This scheme means that the user has different operation levels that need different interface capabilities. First he/she can send a high level task to the planner, so this one generates a trajectory and sends it to the control module, initiating the robot momevent. This is what we call autonomous mode. Second, the user can ignore the planner, and send directly a path to follow to the control program. This may be called semiautonomous mode. And finally, the user can ignore both the planning and control modules, sending velocity commands directly to the mobile robot. This last operation mode is teleoperation.

So, the user interface must allow the operator/user to interact with the robot at any of these levels.

2 Conventional Mobile Robotics Applications

For many years, the use of general-purpose workstations (mainly unix systems) were used to implement user interfaces in robotics. The main reason was that man-machine communication was not considered a key topic for the navigation of a mobile robot, but an auxiliar one [4].

A typical example of mono-robot man-machine interface used for simulation and robot-sensor visualisation with an actual robot is the one that contains in the screen the following information:

- Map of the world with the mobile robot location at any instant.
- Instantaneus values from sensors (typically sonars).
- Filtered sensor data (i.e. local maps).
- Trajectories to be followed.
- Task fo each robot.
- Cells to allow the user task ordering.
- Panels to select the control algorithms.
- A window to draw trajectories with the mouse.
- A window for teleoperating the robot with the mouse.
- etc.

You can find examples for them at any mobile robotics laboratory. On the contrary, and day by day, the feeling that the main problem when searching autonomy was uncertainty gave the idea that perception management representation was extremely important. A good example of this idea comes from the fact that, by using these conventional interfaces, a user encounters a lot of difficulties when teleoperating the robot (by hand), while the control algorithms are able to navigate with a good performance. Does this mean that the sensor information the control modules uses is better that the one the user has in the screen? For sure, yes.

At the same time conventional mobile robot simulators use to be 2D, simulating (usually proximity) sensors in the horizontal plane. There are very few sensors with a 2D behaviour, and examples are not proximity ones.

3 New Perspectives in Autonomous Robots

Experience has shown that the use of conventional xview or MOTIF based systems for MMI is an obsolete technique, and so we must go to 3D flexible environments that integrate on the same screen information from heterogeneous sensors, data fusion with simulated views, as well as direct and processed images from vision systems. Usual applications include simulators, and independent windows/modules for teleoperation, sensor data representation, or autonomous/semiautonomous navigation mode. The transition

is already in progress, with the introduction of specific workstations such as those from Silicon Graphics, and more powerful window systems with 3D capabilities (OpenGL) and programming languages (VRML) [2], which allow to implement virtual reality techniques [5].

The same tool must be able to allow the three functioning modes: simulation, tele-operation and autonomous navigation. The first topic, simulation, is improved by adding the 3rd dimension to each of the sensors. While the mobilr robot kinematics is always a 2D movement, most of the senors enrich their representation: sonars, infrared, cameras, lasers, etc. for example, a sonar may be improved by representing it as a cone instead as a straight line.

Simulation is strongly related with the management of sensor information and so with teleoperation issues. For example, information that the operator needs to teleoperate the mobile robot can increase its quality, by adding lateral views to conventional user interfaces, as a sonar local map (see figure 2). Nevertheless, it has been proved that this kind of 2D information helps, but is not the complete solution to operate remotely the robot. A good and complete solution in present applications includes 3D simulations ovelaped with the real view, visual information from on board cameras, and the posibility to select on-line the image from different points of view.

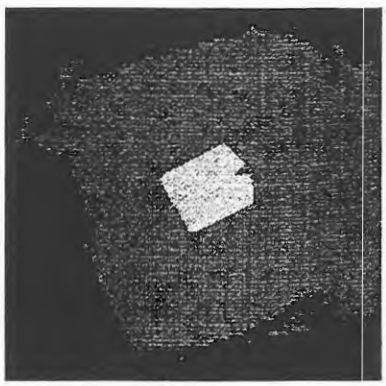


Figure 2: Sonars local map

The third topic is related with autonomy, being the highest level of autonomy a multirobot system were each mobile robot behaves as an holon. THis means that each robot is an autonomous system that is able to take its own decisions but, at the same time, it is part of a greater system. Each robot receives the same tasks and must decide which one to carry out, negotiating, if necessary with the others. Here the user interface includes the communication system as an important part of the system architecture [1]. As an example, we developed a multi-robot application that allows to monitor and send tasks to a set of mobile platforms sharing the space in an indoor environment [3].

Furthermore, general application design has been traditionally done by hand. The use of OOP languages such as C++ and modelling languages such as UML, allow a design that improves the communication and understanding among large software project programmers. Actual trends in this field walk towards the development of tools for the graphical design of the complete system under construction.

The developer also need tools to guarantee an easy testing step of the system. Figure 3 shows the development stages of our multirobot system:

- First, a centralized simulation is done in one computer to test the multirobot coordinated control modules.
- Second, a distributed simulation is done in order to test the speed and feasibility of the communication system, as well as the decision system based on inter-robot negotiaion.
- And third, an actual distributed multirobot system is launched.

Present mobile robotics projects should take advantage of the previous technologies to achieve a better performance.

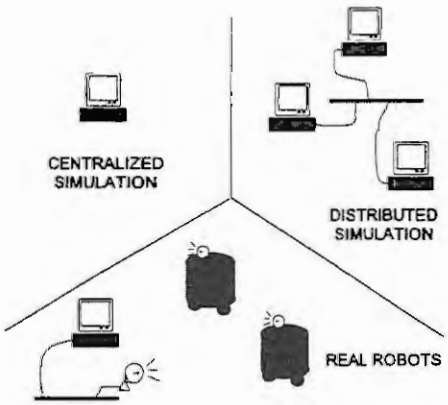


Figure 3: Multirobot system development stages

4 Conclusion

Presnet projects have became, more and more complex software systems that need an appropriate management and co-ordination. At the same time, conventional user interfaces have became obsolete and claim the use of more powerful visualization machines. The present situation is the transtition from conventional user interfaces towards the use of the new available technologies. Nevertheless, a lot of work is still pending.

Acknowledgements. We acknowledge the funding of the Spanish government through the CICYT projects EVS (*Virtual Platform for Autonomous Distributed Systems Engineering*) and M3 (*Modular Intelligent Control*), as well as of the European Community through MOBINET project (Training and Mobility of Researchers program).

References

[1] de Antonio, A., Intelligent Control Architecture. PhD. Thesis, Universidad Politecnica de Madrid (1999).

[2] Cuesta, F., Arrue, B.C. and Ollero, A., A new System for Intelligent Teleoperation of Vehicles, 3rd IFAC Symposium on Intelligent Autonomous Vehicles (1998).

[3] Matia, F., Moraleda, E., Mena, R. and Puente, E.A., Distributed Task Planner for a Set of Holonic Mobile Robots. In Distributed Autonomous Robotic Systems III, Springer-Verlag (1998) 35–44.

[4] Salichs, M.A., Puente, E.A., Moreno, L., Pimentel and J.R., A software Development Environment for Autonomous Mobile Robots. In Recent Trends in Mobile Robots, World Scientifi series in Robotics and Automation 11 (1993) 211–253.

[5] Wexelblat, A., Virtual Reality. Applications and Exploration. AP Profesional (1993).

HUMAN-ROBOT-COOPERATION USING MULTI-AGENT-SYSTEMS

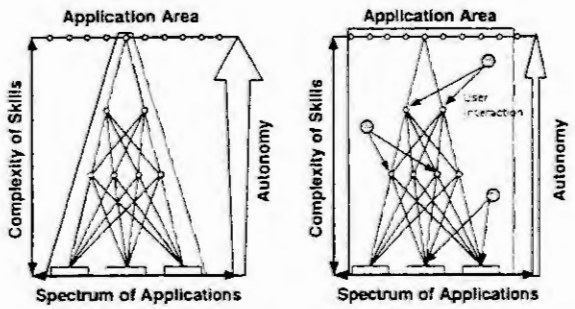
T. Laengle and H. Woern
University of Karlsruhe, Institute for Process Control and Robotics
Kaiserstr. 12, Geb. 40.28, D-76128 Karlsruhe, Germany

Abstract. In this paper an new intelligent robot control scheme is presented which enables a cooperative work of humans and robots through direct contact interaction in a partially known environment. Because of the high flexibility and adaptability, the human-robot cooperation is expected to have a wide range of applications in uncertain environments, not only in future construction and manufacturing industries but also in service branches. A multi-agent control architecture gives an appropriate frame for the flexibility of the human-robot-team.

1 Introduction

Today, the demands for high flexible robotic systems, which must have the capabilities of adapting themselves to an uncertain environment, are rapidly increasing. For example, there is a need of robots for medical surgery, hotel trade, cleaning of rooms or household. Contrary to industrial robots, which are working in a well known environment and used by skill operators, it is not the case for service robots.

In spite of promising researches in the field of artificial intelligence, autonomous robots have still difficulties to execute complex tasks in "turbulent" environment. The more autonomous the system is, the more structured the world model should be, and the more specific its tasks are [3]. A small deviation between the world model and the real world causes the system break down. As a result autonomous robots suffer from typical domain restriction (Fig. 1.a). Up to now it is still not known how to achieve complete autonomous execution of complex tasks in unstructured environment.



a) Intelligent Control in Autonomous Systems

b) Interactive Control in Service Robots

Fig. 1: Domain restriction [5]

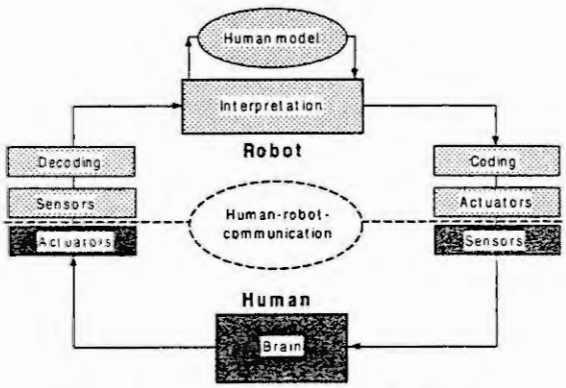


Fig. 2: Interpretation of the information

A possible compromise to avoid the domain restriction problem is to find the right balance between robot autonomy and human-robot-interaction. We don't try to eliminate the uncertainties in the environment nor increase the intelligence of the robot. The main idea is to compensate the uncertainties in order to avoid the robot to become stuck by an appropriate mixture of intelligence, environmental knowledge and human-robot interaction (Fig. Ib).

2 State of the art

In recent years new human-robot interfaces based on direct interaction through contact has been tackled by many researchers. An important issue in this area is the arm-manipulator coordination for the load sharing problem. For the cooperative motion control of a robot and a human the human characteristics are approximated by an impedance model in [2]. In [1] a variable compliance control as a coordination mechanism for the arm and the manipulator is proposed. In [6] a control scheme was proposed to accomplish the coordinated task execution of a human and a mobile manipulator. The majority of these works on arm-manipulator cooperation however focuses only on the

dynamics of the arm-manipulator chain. Hence there is no a priori planning used for the robot, it plays a passive role and it does not recognize or interpret the action of the human according to the current situation or task context.

An other important issue in this area is the shared guidance of an object by the operator and the robot. This problem is addressed especially by the so called "Synergistic devices". Synergistic devices are intended for cooperative physical interaction operator by producing forces. They allow the operator to have control of motion within a particular plane, while the device dictates motion perpendicular to that plane [4].

3 Analysis of the problem

Robots are considered as intelligent autonomous assistant of humans, which can mutually interact on a symbolic (exchange of digital messages on a network) and a physical level (visual, tactile, acoustic). The communication between robots and humans comprises the transmission of the information at the physical level and the interpretation of the information. The transmission of information from one partner to another occurs through predefined interaction patterns, which allow the receiver to understand the intention of the transmitter. Communication transmission of digital data, which enables robots to communicate with each other in normal cases, is however, no longer sufficient in the case of the human-robot-teams. The communication at a physical level, such as contact forces, must also be involved. The sensors should be used not only for the task execution, but also to recognise situations of possible interaction with other agents and to receive the transmitted information as well. The interpretation makes sense out of the transmitted information (Fig. 2). Particularly for the human-robot-cooperation, it is very important to avoid interpretation errors to prevent the human from danger. The transmitter should make sure that his intentions are clear enough to the receiver. For the human the confirmation of the good interpretation of the intention is reached through analysing the implicit behaviour of the partner. But this is possible only if the behaviour of the partner is not ambiguous. Particularly, the robot has to perform human-like motions and to exhibit reflexes similar to those observed in humans: like retracting an arm when it hits something.

4 Realization

Our approach is based on the integration of human-robot cooperation in a task plan. However this plan tolerates some degrees of uncertainty and gives some degrees of liberty to the human, who is able to specify on-line some task parameters and recover system errors. A typical task for a manipulator is a *pick and place task* and can be achieved directly by one operation cycle (Fig. 4). An operation cycle means the following elementary operation (EO) sequence which appears in robot programs frequently and is described in Tab. 1. The correct execution of this operation cycle requires sensing and actuating capabilities as well as a knowledge database. These capabilities are useful in order to sense the state of the dynamic environment, avoid collision with the environment, find the position of the workpiece, grasp the workpiece correctly, compensate the uncertainties about the environment.

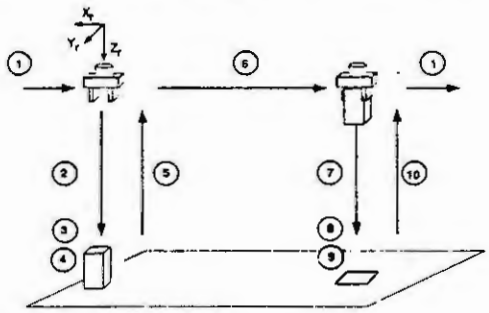


Fig. 3: Representation of an operation cycle

EO 1	Moving an arm without holding a workpiece
EO 2	Approaching the workpiece with the gripper
EO 3	Grasping the workpiece
EO 4	Separating the workpiece from a base plate, a fixture or from another object
EO 5	Departing
EO 6	Moving an arm which holds a workpiece
EO 7	Approaching the workpiece to another object
EO 8	Establishing a defined contact to this object
EO 9	Ungrasping the workpiece
EO 10	Departing

Tab. 1: Description of an operation cycle

In order to integrate the human in the "control loop" of the manipulator we define for each situation of the *pick and place task* symmetrically to the EOs, which are executed automatically, the semi-automatic elementary operations

SEOs. When the robot is not able to accomplish the required task autonomously, it switches into the semi-automatic mode and gives then the operator the possibility to help him. The integration of these semi-automatic SEO into the operation cycle is shown on Fig. 4. We distinguish two levels for the task execution according to the degree of autonomy of the robot: the level A represents the autonomous level S whereas the level represents the semi-autonomous level. The events which activate the transition between the two levels are represented by the white arrows.

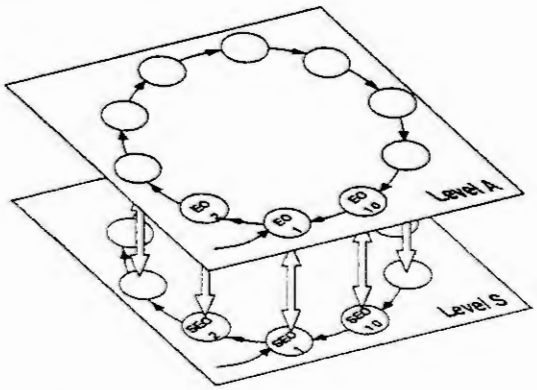


Fig. 4: Integration of the human-robot interaction into the task plan.

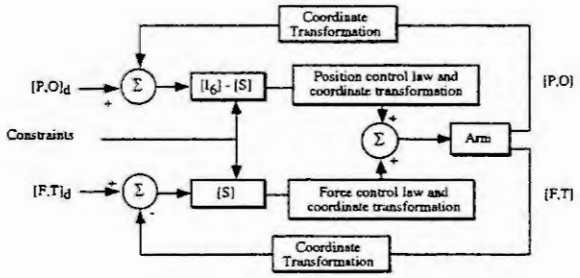


Fig. 5: The hybrid position-force controller

In the semi-autonomous mode the control of the degrees of freedom of the manipulator are shared by the human and the ICS. This is what we call "control allocation". To enable the shared control of a manipulator between the humans and the ICS in the semi-autonomous mode we separate the coordinates into the coordinates controlled by the human and the coordinates controlled by the ICS. To each SEO corresponds a predetermined control allocation. This control allocation is defined according to the capabilities of human and robot. A given degree of freedom is controlled by the "partner" of the human robot team, who has the best sensing, actuating or cognitive capabilities to control it. For example in an unstructured environment it often happens that the robot does not know or does not recognize the place where the piece should be put (EO 6). Thus the position along the X axis and Y axis (see Fig. 3) should be controlled by the human, the robot compensates the weight of the workpiece and controls the orientation.

5 Implementation

For the distributed control of the 6 degrees of freedom of the manipulator among the human and the robot we use a hybrid position-force controller represented on Fig. 5. [P,O] represents the internal states of the manipulator: the position (Px, Py, Pz) and the orientation (Ox, Oy, Oz) of the end-effector. [F,T] represents the external state of the manipulator: The forces (Fx, Fy, Fz) and the torques (Tx, Ty, Tz) applied on the force/torque sensor attached on the end-effector. I_6 represents the 6 dimensional identity matrix whereas S represents the diagonal selection matrix. If a component of S is '0', the corresponding degree of freedom is controlled by the ICS (Position control law), whereas if the component is '1', the degree of freedom is controlled by the human (force control law or compliant motion), see Fig. 6. The cooperative work between human and robot requires not only the shared control of the different DOFs according to the situation, but also the ability to switch to the autonomous mode (Level A) or to the semi-autonomous mode (Level S) at the right time. This mode transition could be initiated whether by the ICS (automatic transition) or by the human (manual transition).

The transition between the level A and S occurs automatically when an error has been detected during the execution of an EO or when the robot recognizes its incapacity to execute the EO because parameters are missing. The transition between the level A and S can also be initiated manually when the operator applies the corresponding forces on the end-effector. All EOs are guarded motion, so that if the applied forces cross a given transition

threshold, the transition is initiated. The transition recognition makes use of rules and facts which tell the system how to react to a given situation. The situation is determined by four types of parameters:

1. The operation cycle status: it characterizes the elementary operation (EO or SEO) which is currently executed.
2. The operation status. It specifies if an error has been detected.
3. The task specification status. It specifies which parameters are missing for the autonomous execution of the task.
4. The sensor data. The sensors are force/torque sensors and position sensors (overhead camera or hand camera).

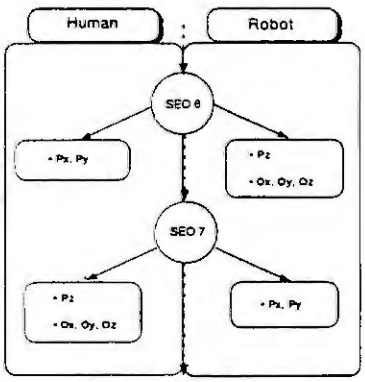


Fig. 6: Control allocation of the degrees of freedom

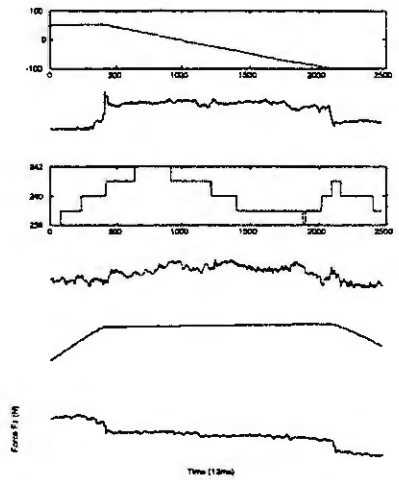


Fig. 7: Position and Forces

6 Acknowledgments

The research work was performed at the Institute for Process Control and Robotics (IPR), Prof. Dr.-Ing H. Wörn and Prof. Dr.-Ing. R. Dillmann, Faculty for Computer Science, University of Karlsruhe.

7 References

[1] Al-Jarrah, O. M.; Zheng, Y. F. (1997): *Arm - Manipulator Coordination for Load Sharing Using Reflexive Motion Control.* IEEE Int. Conf. on Robotics and Automation, pp. 2326-2331, Albuquerque, New Mexico, April 1997.

[2] Ikeura, R.; Monden, H.; Inooka, H. (1994): *Cooperative Motion Control of a Robot and a Human*: Proc. of 3rd IEEE Int. Workshop on Robot and Human Communication (RO-MAN'94), Nagoya, July 18-20, 1994.

[3] Laengle, T.; Rembold, U. (1996): *A Distributed Control Architecture for intelligent Systems.* Int. Conf. on Advanced Sensor and Control-System Interface SPIE, Vol. 2911, pp. 52-61, Boston, USA, Nov. 18-22, 1996.

[4] Peshkin, M.; Colgate, J. E. (1996): *"Cobots" Work with People.* In IEEE Robotics and Automation Magazine, Vol. 3, No. 4, pp.8-9, Dec. 1996.

[5] Rembold, U.; Lueth, T.; Ogasawara, T. From Autonomous Assembly Robot to Service Robots for Factories. IEEE Int. Conf. on Intelligent Robots and Systems (IROS'94), Munich, Germany, Sept. 12-16, 1994.

[6] Yamamoto, Y.; Eda, H.; Yun, X. (1996): *Coordinated Task Execution of a Human and a Mobile Manipulator.* Int. Conf. on Robotics and Automation, Vol.2, pp. 1006-1011, 1996

HUMAN-MACHINE INTERACTION IN INTELLIGENT ROBOTIC SYSTEMS : A UNIFYING CONSIDERATION WITH IMPLEMENTATION EXAMPLES

S.G. TZAFESTAS and E.S. TZAFESTAS
Intelligent Robotics and Automation Laboratory
Department of Electrical and Computer Engineering, NTUA
Zographou, Athens, GR-15773, GREECE

tzafesta, brensham @softlab.ece.ntua.gr
Fax. +30-1-772 2490

Abstract. The goal of this paper is to provide a general unified discussion of the human machine interaction issues as applied to robotics. First, the general structure of intelligent human machine interfaces (HMIs) is presented. Then the class of natural language interfaces (NLIs) is reviewed. Next the class of graphical HMIs (GHMIs) is investigated including the combination of GHMIs with virtual reality (VR) system facilities. Finally, a few implementation examples of HMIs in various robotic systems are briefly outlined.

1. INTRODUCTION

The interaction between a user and a computer or more generally a technological system is performed through the human-machine interface (HMI) which plays a primary role to the effective and successful operation of the system. On average, about 30% of the operational software is needed to support the HMI of a typical knowledge based system. The design of an HMI is a multidisciplinary task needing the cooperation of experts on human cognition, display technologies, graphics, software design, natural language processing, artificial intelligence etc. Actually, the design of an HMI is still more an art than a science. The goal of designing efficient HMI components in robotic, or more generally, in automated systems is to improve operational efficiency and overall productivity while providing a safe, comfortable and satisfying front-end for the operator/user. To this end, the capabilities and limitations of the human operator should be analyzed and used for the HMI design, which involves the tasks, the tools, the software, the environment and the overall organization. In robotics, the need for efficient and human-friendly man-machine interfaces is of predominant importance. However, a general design methodology does not exist, nor a set of generally adopted evaluation criteria.

2. THE GENERAL STRUCTURE OF ROBOTIC HMIs

2.1. General Issues

For a robotic HMI to be intelligent, access to a variety of knowledge sources is required. These include :

- Knowledge of the user
- Knowledge of the user tasks
- Knowledge of the tools
- Knowledge of the domain
- Knowledge of interaction modalities
- Knowledge of how to interact

A good practice for designing a particular intelligent HMI is to require some knowledge in each of the above areas and a lot of knowledge in the areas of particular relevance to the HMI at hand. The general structure of an intelligent HMI is shown in Fig. 1.

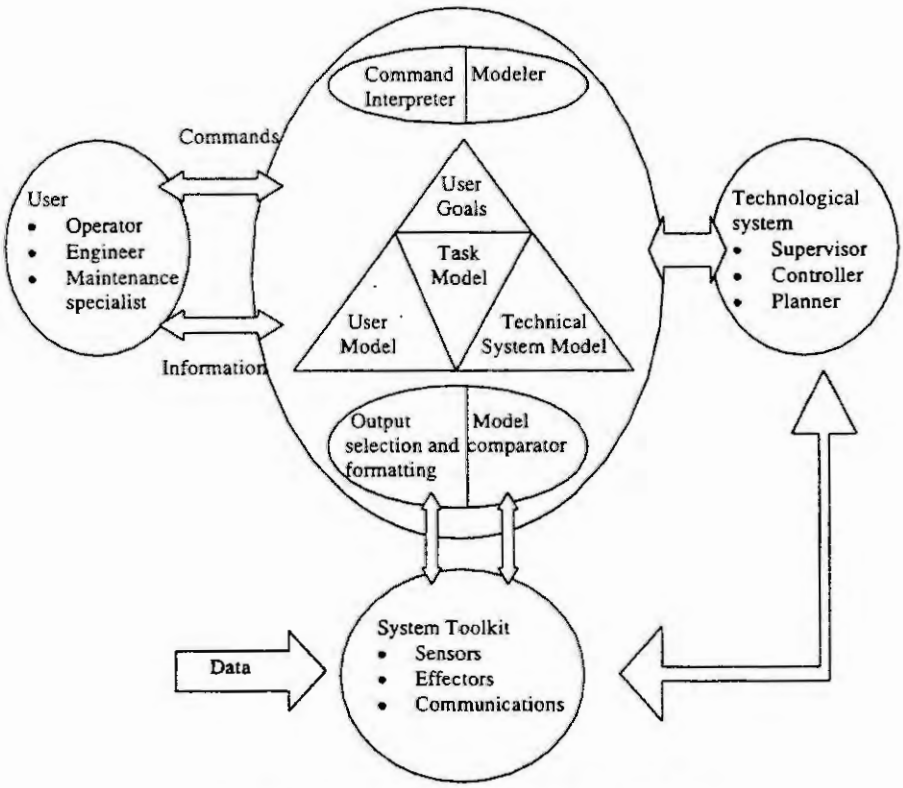


Figure 1. General structure of intelligent HMI

The technological system (robotic, industrial, enterprise, etc.) involves a supervisor, a planner and a controller, and sometimes (depending on its size and complexity) a decision support system (DSS) component which contributes to the realization of cooperative human-machine decision making and control.

The three main types of users are operators, engineers and maintenance specialists. These users interact with the technological system (robotic system, continuous physical/chemical process, manufacturing system, etc.) via the HMI. Users have in general different but overlapping needs with respect to depth and quantity.

2.2. Principal functions of robotic HMIs

The principal functions of robotic HMIs are the following :

- Input handling
- Perception and action
- Dialogue handling
- Tracking interaction
- Explanation
- Output generation

The ***input handling*** function should provide the means to handle the type of inputs received by the system which may be analog, digital, probabilistic, linguistic, fuzzy, etc.

The ***perception and action*** function plays a key role in the overall HMI performance and is supported by the presentation level of the HMI which determines how to present the information to the user and how to transform his/her control inputs.

Dialogue handling (control) deals with determining what information to treat and when. A dialogue is any logically coherent sequence of actions and reactions exchanged between the user and the HMI. Human-machine dialogues are necessary for many robotic operations, e.g. scheduling, supervision, planning, control, etc.

Tracking interaction deals with tracking the entire interaction between the HMI and the human user, as well as between the HMI and the robotic system at hand.

The ***explanation*** function needs a model of the technical system (here the robotic system) to be available. Its role is to explain to the user upon request the meaning of the various aspects and components of the technical system, and sometimes of the HMI itself. It should be also capable of explaining how the various parts of the system operate.

Output generation is realized using graphical editors and typically offers appropriate graphical and textual pictures which are dynamically changing. In more recent applications, multimedia presentations are also provided.

If the HMI is required to be able to adapt to different users or user classes, a ***user model*** is also needed. To design a user model, it is necessary to use our knowledge on human processing behavior and represent the cognitive strategies, via rules, algorithms and reasoning mechanisms. A more complete user model must also include a model of the robotic system in order to incorporate the user's view with respect to the robotic system.

3. NATURAL LANGUAGE HMIs IN ROBOTICS

3.1. General issues

A special class of HMIs, very popular in robotics, is the class of natural language interfaces (NLIs). NLIs possess humanized properties since the user can communicate with the system through a kind of verbal language (e.g. a small subset of English). Actually, NLIs are not the best interfaces in all cases. Thus to decide whether to use a NLI or not, one has to consider several factors, of which some examples are the following :

- ***Cost.*** The cost of NLIs is user higher than that of standard HMIs.
- ***Ease of learning.*** If a full natural language is used, no human effort is necessary to learn it. This is not so if a restricted language with legal statements is used.
- ***Conciseness.*** The desire for conciseness is usually in conflict with the user friendliness.
- ***Precision.*** Many English sentences are ambiguous. This is so natural, that English does not use parentheses as do artificial logical languages.
- ***Need for pictures.*** Words are not the best way to describe shapes, positions, curves, etc. A picture is worth many words. However, programs that handle graphical objects (e.g. CAD systems) are still good candidates for NLIs and other linguistic interfaces.
- ***Semantic complexity.*** Natural languages are concise and efficient when the universe of possible messages is large. Actually, no trivial language can perform the interfacing job, since the number of different messages that have to be handled is extremely large.

The components of a NL understanding system, i.e. a system that transforms statements from the language in which they were made in a program-specific form that initiates appropriate actions, are :

- Words and lexicons
- Grammar and sentence structure
- Semantics and sentence interpretation

Three ways to combine the above primary components into an integrated understanding system are :

Interactive selection. The system displays the options to the user who chooses among them to gradually construct a complete statement, which corresponds to actions that the target program can perform.

Semantic grammars. The window-based approach does not allow the user to control interactions or compose free-form statements that the system has to understand. The alternative is for the user to compose entire statements. A semantic grammar provides one implementation of this alternative approach but is appropriate when a relatively small subset of a NL has to be recognized.

Syntactic grammars. If a large part of NL is used as HMI, the capture of as much of the language regularity as possible is required. To this end, it is necessary to capture the syntactic regularity of the NL at hand. Thus one needs to use a syntactically motivated grammar.

Today, several tools are available to assist in the building process of the lexicon, the grammar, the semantic rules and the code that uses all of them. Also some programs exist that do most of the understanding in all three approaches discussed above.

3.2. Some representative works

NLIs in robotics have been considered and used by many researchers.

- Sondheimer [21] where the spatial reference problem of NL robot control is investigated.
- Nilsson [18], where a mobile robot (SHAKEY) capable of understanding simple NL commands is presented.
- Sato and Hirai [20], where NL instructions are employed for teleoperation control.

- Vere and Bickmore [23], Chapman [3] and Badler et al. [1], where the control of autonomous agents in 2D or 3D work spaces is achieved via NL commands.
- Torrance [22], where a NL interface is used to navigate an indoor mobile robot.
- Neumann [17] and Herzog and Wasinski [9], where the synergetic integration of NL and vision processing in robotic systems is considered.
- Fischer, Buss and Schmidt [5], where a comprehensive NLI is designed and used for a service mobile manipulator (ROMAN).
- Wahlster et al. [24], Bajcsy et al. [2], Neumann [17] and Herzog and Wasinski [9] investigate the utilization of combined sensory information and verbal descriptions in NL interface design for intelligent robotic systems.
- Honig and Vonk [11] consider the use of NLI in robotic fault diagnosis and technical information systems.
- Finally, Koenig [12] examines the issues of knowledge structures and sentence generation in NL communication for interactive man-robot systems. The knowledge structures considered are the extended graph tuple and an extended descriptive relationship. These two knowledge structures are then used as the knowledge suppliers for generating sentences that convey knowledge tuples (KTEs).

4. GRAPHICAL HMIs IN ROBOTICS

The field of graphical HMIs (GHMIs) is very broad. We will only discuss some fundamental issues of GHMIs relevant to robotic systems. In robotics, GHMIs are used for task analysis, online monitoring and direct control. For example, to teleoperate a mobile robot in a critical workspace, a considerable effort must be devoted to preparing the task, training the operator and finding the optimal cooperation modes in various situations. Before actually executing a task, a GHMI can help the user to specify his intention, display the commands and the expected consequences on the monitor. In this way, the user can interactively generate and modify a plan.

On a GHMI an operator can define a series of movements and actions by clicking or dragging a mouse on the screen. The available task and geometric planners can then find a sequence of motions and actions that implement the task. A simulation system is usually designed and used to animate the robot's motion on a 2D or 3D workspace, where several viewpoints can be set to monitor and observe the robot's behavior and its relation to the world. Possible collisions with obstacles, robots and other objects are avoided. Here the optimal utilization of various sensors is a basic requirement. As an additional aid, a ***task editor*** is necessary to support the task specification by interactively modifying a plan. It is useful if with this task editor the operator can also define a sequence of actions as a macro. The macros can be retrieved and used to represent and implement an entire task plan. A useful concept that can be used in task analysis is the concept of ***telesensor programming*** (Hirzinger, [10]). Due to the unavoidable errors in the dead-reckoning and world models, the sensor patterns have to be employed by the robot to ensure an accurate relation with the world.

Graphical interfaces are frequently combined with animation and virtual reality (VR) tools. Examples of this type are the works of Rossmann [19], Wang et al. [25] and Heinzmann [8].

In [19] the multirobot system (called CIROS) implements the capability to derive robot operations from tasks performed in the virtual reality environment. To this end, two appropriate components are used, the change-detection component and the change-interpretation component. The VR system employed is based on a special simulation system (COSIMIR : Cell Oriented SIMulation of Industrial Robots) developed at the Institute of Robotics Research in Dortmund, Germany. In CIROS, a new VR concept is used. This is called ***projective virtual reality*** (PVR), because the actions carried out by humans in the VR are projected on to robots to carry out the task in the physical environment. The intelligent controller implements PVR-based control by adding the levels for ***online collision avoidance***, multirobot coordination and automatic action planning.

The following requirements have been met in the CIROS system :

- Time delays between the display of a robot's movement in the VR environment and its physical movements
- Accurate graphical modeling
- Very precise data glove
- Reduction of the "trembling" of the operator's hand
- Online collision avoidance
- Versatile sensor control to compensate for undesired tensions when objects are inserted into tight fittings.

In [25] the human-machine system includes a virtual tools system, an automatic path planner and a collision detection simulator. Tests on the performance of the path planner are also discussed. A virtual tools HMI for point-

specification of tasks, which interweaves virtual robot end-effector representations with physical reality to immerse the human in the scene using simple hand gestures is developed for flexibly designating where the robot should grasp as an incoming part. The virtual tools system is displayed in four quadrants on a Silicon Graphics workstation with Galileo video. The virtual gripper is displayed on the two left quadrants display, superimposed on two camera views and blended with live video, to create the illusion of a real gripper in two views in the physical scene. The top right quadrant is occupied by the toolbox of graphic icons representing various tools available for use by the robot. The bottom right quadrant displays homogeneous transformation matrix information such as graphic object models, views from the robot camera, etc. This Pennsylvania University system which is based on the virtual tool concept allows the operator to direct robot tasks in a natural way in almost real-time.

Finally, in [8] the HMI of the robot consists of a visual face tracking system. The system employs a monocular camera and a hardware vision system to track several facial features (eyes, eye brows, ears, mouth, etc.). The 3D pose and orientation of the head is computed using this information. The paper provides a solution to the design of human-friendly robots by satisfying two safety goals.

Safety goal 1 : A human-friendly robot should be able to operate without posing a threat when humans are inside the robot's workspace.

Safety goal 2 : In an unstructured environment which may involve humans, any action autonomously taken by the robot must be safe even when the robot's sensor information about the environment is uncertain or false.

5. SOME EXAMPLES

5.1. The ROMAN NL HMI

The ROMAN service robot system was designed and built in the Munich Technical University ([7]). In this system, the information exchange between the operator and the robot is performed in two stages; task specification and task execution (semi- or fully autonomous). The task specification requirements include task description. The task execution requirements involve approaching the goal area, object specification, object handling and object hand-over. In addition, there are the monitoring and sensor support requirements. The human robot dialogue involves :

- Dialogue-oriented natural speech (voice) command input
- Visual screen-based monitoring
- Tactile supervisory control during mobile handling
- Voice output during task operation

The NL HMI architecture of ROMAN is as shown in Fig. 2.

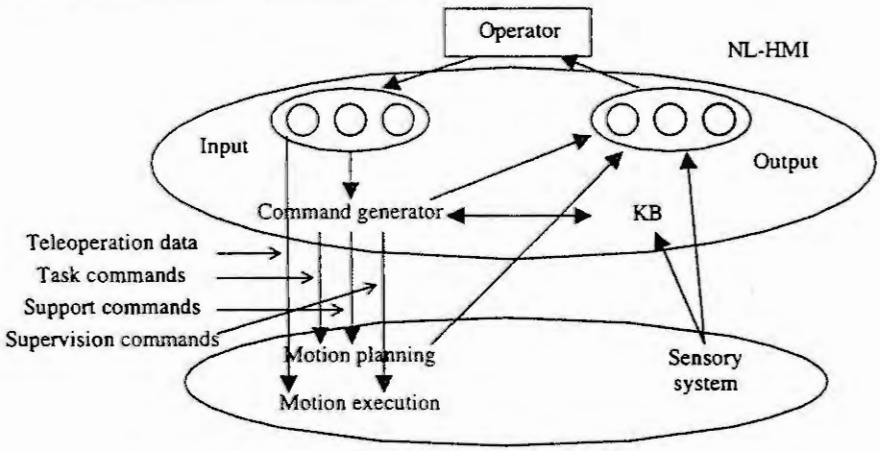


Figure 2. NL-HMI architecture of the ROMAN service robot

The ***task commands*** consist of service task-specific actions, the ***support commands*** are the operators' responses to requests received from the motion planning level during task execution and the ***supervision commands*** are initiated by the operator during task execution and immediately interrupt the current operation. The command language is able to represent both the user-defined service tasks and service robot-specific commands ([5][6]).

The sensor information passed to the NL-HMI from the planning level involves : off-line environmental data, continuous sensor data and abstract sensor data. Any problem arising during task execution initiates a request for support, which needs to be interpreted by the human operator.

The command generator translates semantic structures into robot commands. The command generator of ROMAN receives the operator's instructions and performs the following functions :

- Translation
- Consistency check
- Completeness check
- Data expansion
- Macro separation

Its output is the corresponding robot command

5.2. The KAMRO NL HMI

The KAMRO autonomous mobile robot was designed and built in the University of Karlsruhe ([13][16][14]) and uses a multi-agent architecture.

A basic problem in such multi-agent systems (MAS) is the negotiation among the agents that compete for a given task. This negotiation process can be performed by a *centralized mediator* or a *selected candidate* or by *many (or all) candidates*. All agents should be able to negotiate with the competing agents. One way to manage the communication among agents is via a blackboard system. Possible deadlocks in a MAS are the following :

- Deadlocks caused by agent bodies/external resources
- Deadlocks caused by special agents
- Deadlocks caused by agent teams

These deadlock situations are translated into corresponding mechanisms in the KAMRO robot.

The NL HMI of KAMRO performs the following functions :

- Task specification/representation, i.e., analysis of instructions related to the implicit robot operations (e.g. "pick-and-place")
- Execution representation
- Explanation of error recovery
- Updating and describing the environment representation

The KAMRO NL HMI architecture is as shown in Fig. 3.

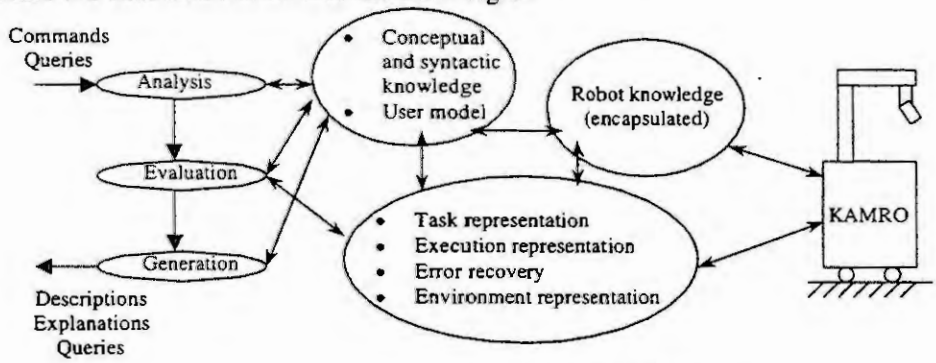


Figure 3. NL-HMI architecture of the KAMRO robot

The robot and the NL HMI have permanent access to the correct environment representation via an overhead camera. This information is stored in a common database. Since the world representation changes over time, a timestamp of the snapshot is used which allows merging older and newer knowledge about the environment.

The processing of NL instructions is illustrated in Fig. 4.

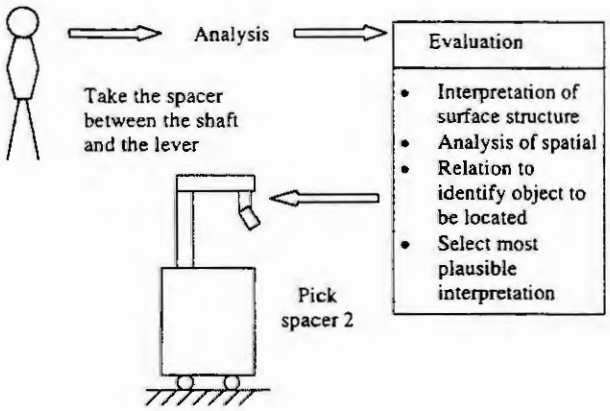


Figure 4. Structure of NL instruction processing

5.3. The PRIAMOS GHMI

A GHMI was employed in the Karlsruhe PRIAMOS robot ([4][15]).

PRIAMOS design has taken into account the following :

- Degraded perception from the vision feedback at the operator site and time-delay between operator command and robot execution.
- Provision for several control modes to facilitate different degrees of cooperation between operator and robot.
- Continuous learning (knowledge accumulation) to allow the robot to gain more autonomy after the guidance of the operator.
- Provision for a GHMI and animation system for task specification, visual display of sensory information and monitoring.

A multisensory system was embodied to the PRIAMOS robot consisting of: (i) a ring of 24 ultrasonic sensors, (ii) an active vision system, and (iii) diode lasers with special optics mounted on the two front corners of the robot. The laser light can be distinguished from the light of surrounding sources via an interference filter (featuring a transmission peak corresponding to the wave length of the laser light) provided to the camera.

A Silicon Graphics workstation is used for monitoring the PRIAMOS robot with sensory feedback in 3D animation space. PRIAMOS was directly controlled to move around the laboratory. The status of the robot and its sensor values are monitored from within the GHMI. Even if the cameras mounted on the robot provided different views of the environment, they could not cover the entire scene. The robot and its world are displayed on the 3D graphics using the "Kavis" animation system. As the robot moves around a building with rooms and walls, the operator has to guide the robot along the wall to monitor/explore the environment. Shared control consists of the robot keeping its distance and orientation with respect to a side wall while being remotely commanded to move ahead.

REFERENCES

[1] Badler, N.I., B.L. Webber, J. Kalita and J. Esakov, Animation from instructions, in: N.I. Badler (ed.), Making them move: Mechanics, control and animation of articulated figures, Kaufmann, San Mateo, CA, 1991, pp. 51-93.

[2] Bajcsy, R., A. Joshi, E. Krotkov and A. Landscan, A natural language and computer vision system for analyzing aerial images, Proc. 9th IJCAI, Los Angeles, CA, 1985, pp. 919-921.

[3] Chapman, D., Vision, instruction and action, MIT Press, Cambridge, MA, 1991.

[4] Dillmann, R., J. Kreuzinger anf F. Wallner, The control architecture of the mobile robot PRIAMOS, Proc. 1st IFAC Intl. Workshop on Intelligent Autonomous Vehicles, 1993.

[5] Fischer, C., M. Buss and G. Schmidt, Human-robot interface for intelligent service robot assistance, Proc. IEEE Intl. Workshop on robot and human communication (ROMAN), Tsukuba, Japan, 1996, pp. 177-182.

[6] Fischer, C., M. Buss and G. Schmidt, Hierarchical supervisory control of service robots using human-robot interface, Proc. Int. Conf. On Robots and Systems (IROS), Osaka, Japan, 1996, pp. 1408-1416.

[7] Fischer, C. and G. Schmidt, Multi-modal human-robot interface for interaction with a remotely operating mobile service robot, Advanced Robotics, Vol. 12, No. 4, 1998, pp. 397-409.
[8] Heinzmann, J., A safe control paradigm for human-robot interaction, J. of Intelligent and Robotic Systems, Vol. 25, 1999, pp. 295-310.
[9] Herzog, G. and P. Wasinski, Visual translator Linking perceptions and natural language descriptions, Artificial Intelligence Review, Vol. 9, 1994.
[10] Hirzinger, G., Multisensory shared autonomy and telesensor programming : Key issues in space robotics, Robotics and Autonomous Systems, Vol. 11, 1993, pp. 141-162.
[11] Honig, J. and A. Vonk, Natural language and technical information systems, Tech. Report ESPRIT CMSO Project, TU Delft, 1989.
[12] Koenig, C., Natural language communication in interactive man-machine systems : Knowledge structures and the generation of sentences, in: S.G. Tzafestas (ed.), Knowledge based systems : Advanced concepts, techniques and applications, World Scientific, Singapore, 1997, pp. 293-318.
[13] Laengle, T. and U. Remboldt, Distributed control architecture for intelligent systems, Proc. Intl. Symposium on Intelligent Systems and Advanced Manufacturing, Boston, MA, USA, 1996.
[14] Laengle, T., T.C. Lueth, U. Remboldt and H. Woern, A distributed control architecture for autonomous mobile robots – Implementation of the Karlsruhe multi-agent robot architecture, Advanced Robotics, Vol. 12, No. 4, 1998, pp. 411-431.
[15] Lin, I.-Shen, F. Wallner and R. Dillmann, An advanced telerobotic control system for a mobile robot with multisensor feedback, in: U. Remboldt et al. (eds.), Intelligent Autonomous Systems, IOS Press, 1995, pp. 365-372.
[16] Lueth, T.C., and T. Laengle, Task description, decomposition and allocation in distributed autonomous multi-agent robot system, Proc. IEEE/RSJ Intl. Conf. On Robots and Systems (IROS'94), Munich, Germany, 1994, pp. 1516-1523.
[17] Neumann, B., Natural language description of time-varying scenes, in: D.L. Waltz 9ed.): Semantic structures, Lawrence Erlbaum, Hillsdale, NJ, 1989, pp. 167-207.
[18] Nilsson, N.J., Shakey the robot, Tech. Note No. 323, AI Center, SRI International, Menlo Park, CA, 1984.
[19] Rossmann, J., Virtual Reality as a control and supervision tool for autonomous systems, in: U. Remboldt et al. (eds.), Intelligent Autonomous Systems, IOS Press, 1995, pp. 344-351.
[20] Sato, T. and S. Hirai, Language-aided robotic teleoperation system (LARTS) for advanced teleoperation, IEEE J. Robotics and Automation, Vol. 3, No. 5, 1987, pp. 476-480.
[21] Sondheimer, N.K., Spatial reference and natural language machine control, Int. . Man-Machine Studies, Vol. 8, 1976, pp. 329-336.
[22] Torrance, M.C., Natural communication with robots, M.Sc. Thesis, DEEC, MIT Press, MA, 1994.
[23] Vere, S., and T. Bickmore, A basic agent, Computational intelligence, Vol. 6, No. 1, 1990, pp. 41-60.
[24] Wahlster, W., H. Marburger, H. Jameson and A. Busemann, Over-answering Yes-No questions : Extended responses in a NL interface to a vision system, Proc. 8th IJCAI, Karlsruhe, Germany, 1983, pp. 643-646.
[25] Wang, C., H. Ma and D.J. Cannon, Human-machine collaboration in robotics : Integrating virtual tools with a collision avoidance concept using conglomerates of spheres, J. of Intelligent and Robotic Systems, Vol. 18, 1997, pp. 367-397.

GRASP PLANNING FOR THREE-FINGERED DEXTROUS HANDS IN VR

Ervin Tóth
Technical University of Budapest
Pázmány Péter stny. 1/D, room 313; H–1117 Budapest, Hungary; ervin@sch.bme.hu

Abstract. More and more robot control systems require intelligent extensions such as virtual reality (VR) in order to be capable of easy and flexible interaction with the environment or the humans. This paper describes virtual reality based programming of the experimental control system of the Puma 560 robot and the dextrous hand developed at the Technical University of Budapest. With the use of the detailed graphic models of the VR system, efficient off-line robot programming and simulation is available. A two-level robot programming language, which was written for this robot-hand system, is introduced. It includes a grasp planning algorithm, which tries to find a prehensile grasp on the object. The planner takes the geometric and material properties of the object into account, e.g. frictional coefficient, parts of its surface that can be touched, and orientation constraints. Furthermore, it considers the obstacles around the object with respect to the position of the robot arm before grasping the object.
Keywords: virtual reality, robot programming, grasp planning.

Introduction

Several robot programming languages has been created during the development of robots. Because of the versatile requirements, an all-purpose language is very unlikely to be widely used. At our institute a two-level language called SRPS (Simple Robot Programming System) has been developed, specially for a NOKIA-Puma 560 industrial robot equipped with a TUB-PC three-fingered dextrous hand. The most charasteristic feature of the language is that it has two levels: a high, task-oriented level and a low, motion-oriented one. On the higher level indirect commands exist, such as 'Grasp a given object with respect to some constraints' or 'Move the dextrous hand above a given object without colliding with the environment'. The language is upwardly compatible, that is, on the high level all the instructions of the low level can be used. The high-level part, naturally, contains commands which require sophisticated algorithms. Two of them has been addressed, namely the grasp planning and the motion planning in the presence of obstacles. This paper demonstrates the grasp planning algorithm in detail.

The VR environment

The virtual reality is used to visualize the actions of the robot for the human operators and this usually has to be done realtime. It is also a requirement to determine that the robot can do the desired action, so that it does not clash with itself and/or the surrounding items of the environment. Hence an efficient multi-level collision-detection algorithm was implemented.

There are two fundamentally different ways to view the robot during the visualization: it can be an overlaid image with the real camera images or can be a view from an arbitrary viewpoint. In order to put the virtual world together with the real environment, the calibration of the system is required. It means that the parameters of a camera, e.g. position, orientation, focal length, etc. are identified based on pictures taken by a real camera, and used in the visualization stage [1]. The system runs on a Windows NT workstation and was developed using the OpenGL graphic library for realtime display and Visual C for all the other computations.

Our system uses boundary representation (B–Rep) scheme to describe objects. For grasp computations the fingertips are treated as a sphere.

The normal vector of the surface element touched by the fingertip is, however, not necessarily equals to the normal of the triangle. For curved objects vertex normals are defined, which are the weighted sum

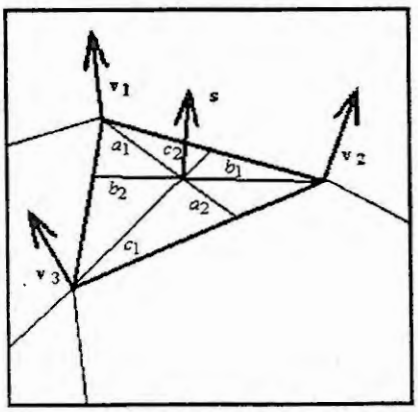


Fig. 1. Surface normal for curved objects

of the triangle normals in a vertex shared by the neighbouring triangles. The weighting factors are the areas of the triangles (the vertex normals are not of unit length). The computed normal of a surface element of a triangle is the linear combination of the three surrounding vertex normals. With the notations of Fig. 1. $\mathbf{v}_i$ are the vertex normals, s is the surface element normal to be computed, a_i, b_i, c_i are distances of the inspected point from the vertices and edges along the line between a vertex and the surface point. Now $\mathbf{s}' = \frac{\mathbf{v}_1 \cdot a_2}{a_1 + a_2} + \frac{\mathbf{v}_2 \cdot b_2}{b_1 + b_2} + \frac{\mathbf{v}_3 \cdot c_2}{c_1 + c_2}$, $\mathbf{s} = \frac{\mathbf{s}'}{|\mathbf{s}'|}$, which means that the vertex normals are linearly interpolated and normalized along the surface.

Contact force computation for simulation is based on the penetration of the fingertip model to the object model. Let the penetration vector, which is parallel with the surface element normal and points from the surface to the deepest point of the finger, be **p**. The contact force is $\mathbf{F} = -\left(K_c + B_c \cdot \frac{d\mathbf{p}}{dt}\right) \cdot \sqrt{|\mathbf{p}|} \cdot \mathbf{s}$, where K_c specifies the nonlinear stiffness of the material and B_c is the damping [2].

The SRPS language

Our goal was to develop a robot programming language which is capable of describing complex motions, e.g. the robot should be controlled either in joint coordinates and as a function *f(x,y,z,t)* too. If the system is used without external information, e. g. for simulation, the contact forces have to be simulated. In our case it means collision and minimum–distance computations between the model of the dextrous hand and the surrounding objects. The language is a frame system, which means that certain actions do not need to be directly defined, only the starting and ending positions and the path and the grasp planner will do the rest automatically.

Some patrs of the functions implemented in the SRPS can be found in the ARPS, the programming language of the NOKIA Puma robot. Certain programs written in SRPS can (with minor modifications) be transmitted to the NOKIA Puma–560 ARPS interpreter. The new functions mainly deal with the control of the robot hand (high–level grasp synthesis) because the ARPS is not applicable for this task. Since the robot carries items during its movement, condition–systems had to be developed which determine if the hand holds the object or not in a given configuration.

With the graphic model the reference points of the robot track can be generated. It's especially important when the robot can't move free because the environment contains obstacles. The virtual robot can be positioned near the desired end positions, then collision–free configurations must be manually found. From this point the path planning algorithm can find the movement between the end points so the operator doesn't have to find and teach it. In SRPS level this means that there are indirect moving commands between endpoints. The output of the path planning is the complete SRPS program, which contains simple (direct) instructions only, which can be directly transmitted to the robot controller. The reference points in both stages are collision–free of course.

To demonstrate the possibilities of the SRPS, a few low-level functions are listed below.

LOAD – used to load predefined coordinates. These are the Denavit–Hartenberg joint (wrist) coordinates. In the SRPS program a coordiante–configuration record is referred with its number.

FRAME/UNFRAME – defines/deletes a new coordinate system in which the following commands work. This is used for navigation close to the manipulatable objects, using their own coordinate system.

GO, GOS – moves the robot between two reference points. With *GO* the path planning is done in joint coordiantes, with *GOS* it is done along a straight line in 3D. The starting position is the actual position of the robot, the end coordinate is given with its number. If a collision is found during the movement, the program stops with an error message.

GRASP – directs the hand into one of the loaded hand positions. If a segment of a finger collides with an object, the phalanx stops. The other phalanxes and fingers moves further until another collision or the desired position. The command has several parameters, such as maximum contact forces, and list of the segments which can touch the object. There are three simplified versions of this command, such as *PINCH, SNAP* and *GRIP* [3]. These commands move the fingers into predefined positions, so these do not need to be taught. Furthermore, these have much smaller number of parameters.

OPEN – straightens the fingers and translates them to the edge of the palm. During the movement collision must not occur. (If so, the program shows an error message.)

The high-level functions:

GOIN (Go Indirect) – this command is not interpreted by the ARPS but is replaced by the path planning algorithm with series of GOs and GOS'.

GRASPIN (Grasp indirect) - directs the grasp planner to develop a grasp. The grasp planning algorithm tries to find a prehensile grasp on the object. It takes the properties of the object into account, e.g. frictional coefficient, center of gravity, parts of its surface that can be touched. Furthermore, it considers the obstales around the object with respect to the initial and final position of the robot arm before approaching and after grasping the object.

Grasp generation

The theory of grasp planning is also studied for a long time. Generally, the object to be grasped has arbitrary shape, given in a form of geometric model, transformed from a modeller software (CAD) or derived from sensor data. Because of the big amount of data required to describe even a medium-complex object, direct (closed-form) solution for finding the optimal grasp does not exist. To overcome this, we assume that using heuristics to locate contact points on the object, a number of grasp candidates can be found, of which the best will be used. Precision grasps are considered (only the figertpis, which are hemisphere-shaped, touch the object). Unfortunately it can not be told how our best candidate is close to the optimal grasp. However, a number of measures to qualify grasps has been proposed, e.g. [4], which computes the external forces that can be balanced by a grasp. From this point of view, the best ones are the form-closure grasps, which resist all the external forces and torques. This induces a special problem: the TUB-PC hand has three fingers, and a 3D form-closure grasp require at least four contact points. The solution is to find a concave corner on the object, which can provide more than one contact point for a single fingertip. It should be noted that not all the surfaces are necessarily included to the grasp search. Individual surface elements can be excluded for some reason, e.g. if a mug is filled with liquid, the wet surfaces would be excluded.

Our contact point generator has two fundamentally different algorithms, one for form-closure grasp generation, the other for arbitrary grasps. The planner decides whether a form-closure grasp is possible, and if not, runs the second algorithm. First, concave corners are searched for. A corner (vertex) is concave, if all the edges meeting in the vertex are on the same side of a plane, and the plane, in the vicinity of the vertex, lays inside the object. The resulting grasps can be divided into three categories: a) which contain only concave corners as grasp points, b) which contain no concave corners, c) mixed. The latter ones can be either form or force closures. Only those concave vertices are considered which are accessible for a fingertip. Next, the vertex normals of the concave corners are searched for triplets $\mathbf{v}_1$, $\mathbf{v}_2$, $\mathbf{v}_3$, where ($\mathbf{v}_1$ *dot* $\mathbf{v}_2$),($\mathbf{v}_2$ *dot* $\mathbf{v}_3$), ($\mathbf{v}_3$ *dot* $\mathbf{v}_1$) are all close to –0.5, which means that these vectors span a quasi-regular triangle. (The triplets are assigned a value w: $\sum_{i=1,2,3;\, j=2,3,1} \left(-0.5-(\mathbf{v}_i \; dot \; \mathbf{v}_j)\right)^2$ and are sorted in ascending order.) If the smallest w is larger than a given w_{max} value, the algorithm fails. We found that 1 is a reasonable value for w_{max}. The remaining ones ($w < w_{max}$) become grasp candidates.The second algorithm tries to place a Y-shaped 'grasp star' into the object that the penetration points on the object surfaces are the contact points. First, a surface is picked. Through its center point **P** a ray is shot inside the object which is parallel to the surface normal. Let its farthest penetration point be **P'**. On the **PP'** line a point **C** is picked; C is as close to the center of gravity of the object as possible. If C lies outside the **PP'** interval, C is chosen that **PC** = 2**P'C**. From C two rays are shot, their intersection point with the object is **Q** and **R** (the angles between **CP**, **CQ** and **CR** are 120 degrees, see Fig. 2.). At this point, we constructed a three-tip star whose tips are the contact points. The algorithm checks whether the **CP**, **CQ** and **CR** lines are inside the corresponding friction cones. If so, a grasp candidate is ready; if not, we try to compensate with altering the star. There are a number of possibilities: the **C** point can be altered, the angles of the star can be modified, and the star can be rotated around the **CP**, **CQ** and **CR** lines. These operations take the surface normal in the contact points into account so that the accumulated deviation from the surface normals becomes smaller. Random modification is also applied, thus the algorithm resembles to the simulated annealing optimization method. During the algorithm, the following criteria are checked: a) two contact points can not be too close to each other, b) the contact points must be reachable for the hand, c) a contact point can not be too close to an edge.

We generate several grasp candidates. If possible, the starting position not only a surface point, but a vertex normal of a concave corner, resulting a mixed type grasp. Finally, the palm position of the dextrous hand is calculated for the grasp candidates. The palm position is selected so that the accumulated distance of the palm from the surrounding objects is maximal.

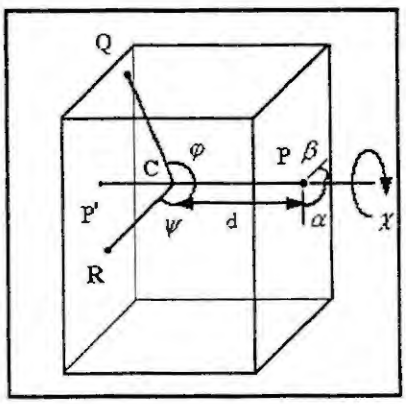


Fig. 2. Parameters that alter the grasp star

Grasp evaluation

The necessary and sufficient conditions of stable grasp are checked. The static equilibrium of fingertip forces and Coulomb friction is considered. Yoshikawa et al. have presented necessasry and sufficient conditions [5]: a) total moment of fingertip forces is zero, b) total force of fingertip forces is zero, c) each fingertip force is in the friction cone. The first condition is satisfied by selecting the fingertip forces so that they intersect in **C**. The necessary and sufficient condition for the second is: $e_i^T(e_j+e_k)\le 0$; $e_j^T(e_i+e_k)\le 0$; $e_k^T(e_i+e_j)\le 0$, where e_i is the unit vector of the i^{th} fingertip force. The third condition is a restriction about Coulomb friction.

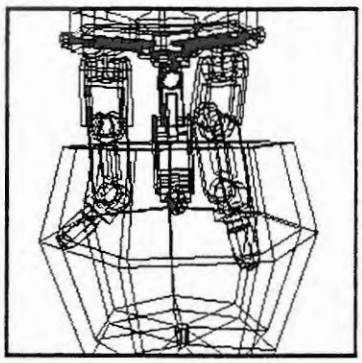


Fig. 3. Internal type form closure

The method we use, proposed in [6], calculates a measure by determinig the set of external wrenches (grasp wrench space, *GWS*) that can be resisted by distributing one unit force over all grasp points. For linear computations, the set of forces within the friction cones at contact point i is approximated by a linear combination of a finite set of n unit force vectors $f_{i,j}$ at the friction cone boundaries: $f_i = \sum_{j=1}^{n} \alpha_{i,j} \cdot f_{i,j}$, $\alpha_{i,j} > 0$, $\sum_{i=1}^{n}\sum_{j=1}^{k} \alpha_{i,j} \le 1$. The resulting generalized force (wrench) at the i^{th} contact can be expressed as $w_i = \sum_{j=1}^{n} \alpha_{i,j} \cdot w_{i,j}, w_{i,j} = \begin{pmatrix} f_{i,j} \\ \lambda \cdot (r_i \times f_{i,j}) \end{pmatrix}_j$. The grasp wrench space can be calculated as $GWS = \text{ConvexHull}\left(\bigcup_{i=1}^{n}\{w_{i,1},\ldots,w_{i,m}\}\right)_j$. The time-consuming calculation of the convex hull can be sped up applying incremental calculations [6]. To establish whether a grasp is a form closure, the following condition is checked [7]: if the origin of the wrench space (R^6) lies exactly inside the convex hull of the primitive contact wrenches w_i, then the grasp is a form closure, like the one depicted on Fig. 3.

Summary

This paper proposed a method for finding and evaluating stable grasps on arbitrary shaped objects. Extended polygonal models supplied by material properties are applicable for contact force simulation. A two-level robot programming language has been introduced to describe complex operations. As a part of this language, a grasp planning algorithm was shown, which generates many grasp candidates based on a heuristic search. Finally, an evaluation function measures the stability of the grasps.

Acknowledgement

Support for the research of grasp planning in virtual reality is provided by the Hungarian National Research programs under grant No. FKFP 0417/1997 and OTKA T 029072.

References

1. Tóth, E. and Tél, F., Intelligent Robot Control System with Graphic Model Based Programming and Stereo Vision. In: Proc. IEEE Int. Conference on Intelligent Engineering Systems, Slovakia, 1999, 57-62.
2. Maekawa, H. and Hollerbach, J. M., Haptic Display for Object Grasping and Manipulating in Virtual Environment. In: Proc. IEEE Int. Conference on Robotics & Automation, Leuven, Belgium, 1998.
3. Lantos, B., Some Possibilities to Increase the Intelligence in Robot Control Systems. Proc. IEEE Conference on Intelligent Engineering Systems, Vienna, Austria, 1998, 7-18.
4. Ferrari, C. and Canny, J., Planning Optimal Grasps. In: Proc. IEEE Int. Conference on Robotics & Automation, Nice, France, 1992, 2290-2295.
5. Yoshikawa, T. and Nagai, K., Manipulating and Grasping Forces in Manipulation by Multifingered Robot Hands. IEEE Trans. on Robotics and Automation, 1, (1991), 67-77.
6. Borst, C., Fischer, M. and Hirzinger, G., A Fast and Robust Grasp Planner for Arbitrary 3D Objects. In: Proc. IEEE Int. Conference on Robotics & Automation, Detroit, Michigan, 1999, 1890-1896.
7. Mishra, B., Schwartz, J. T., Sharir, M., On the Existence and Synthesis of Multifinger Positive Grips. Algorithmica, Special Issue: Robotics, vol. 2, 4, (1987), 541-58.

SIMULATION METHODS FOR MANUFACTURING SYSTEMS DEVELOPMENT

Gunnar Bolmsjö, Lars Randell, Lars Holst and Ulf Lorentzon
Department of Mechanical Engineering, Division of Robotics, Lund University, Sweden

Abstract. In traditional product development, the phases *Market*, *Design* and *Production* are almost sequential activities with only little overlap. In practice, the development process is an iterative activity, but the statement that it can be seen as a sequential process is based on the long iteration loop before any real feed-back information can be utilized in the development process. However, to shorten time to market it is necessary to cut such sequential activities to a minimum and introduce methods and tools which provide possibilities for a parallel *process development* in design, production and market. What will be discussed in this paper is methods and software simulation tools to integrate information between the three main activities *Market*, *Product development* and *Production development* with a focus on the manufacturing system that address issues related to human interaction, large simulation models and concurrent development.

1. Introduction

Rapid product development which can be seen today as a result of the present market orientation put great demand on concurrent engineering on every level in the development process. In the context of product development, the term *process* should be seen as the total activity needed to create a product for the market. It is obvious that such a process must include the aspects of designing a product, but an efficient and competitive product on the market must also fulfill all user oriented functional specifications as well as production requirements related to quality and price. This can in summary be illustrated as in figure 1.

By tradition, product development processes involves a high degree of CAE[1] tools including CAD[2] modeling tools, structure analysis tools and other specialized analysis software. During this process, physical prototypes and mock-ups may also be built for specific testing such as validating the design and market need [2, 3]. However, there are very few possibilities during this process to prepare and validate the production system, besides NC-code preparation for a NC-machine. Assembly of a product can be analyzed with respect to packaging problems using CAD tools, but the actual assembly operations as made by human workers are a complex task to verify by traditional methods. For specific operations in the production of components to a new product, such as arc welding, specific analysis and simulation of a robot arc welding cell can provide valuable information related to design such as weld joint shape, accessibility for the weld gun, etc [4, 5]. Putting operations together into a production system is important as it in an early stage creates a basis for evaluating possible scenarios related to production volumes, logistics and dynamics on the shopfloor. By this, planning the production of a new product can benefit from resources elsewhere in the factory or better utilization of equipment when analyzing the total facilities of the manufacturing system [6].

The simulation tools selected for the project are all based on Deneb Robotics Products *Envision TR* and *QUEST*. *Envision TR* is a time continuous simulator typically used as a simulation tool in robotics with open architecture to allow for integrating software modules for model based process simulation. QUEST is an interactive 3D graphical discrete event simulator with advanced facilities to integrate simulation models with planning tools and PLC systems. This provide a useful platform to further develop methods to model the factory with higher level of detail which is important to obtain the required precision in planning virtual production systems.

2. Integrated product and manufacturing systems development — A case study

The research presented is done in close collaboration with a project group within BT Product AB. This group is responsible for introducing simulation techniques to the development process and consist of engineering staff from different areas including market, design and production. Connected to this project group, a research project was formed with research staff from IVF[3] and Lund University, Robotics Group. By connecting experts from different

[1] Computer Aided Engineering
[2] Computer Aided Design
[3] The Swedish Institute of Production Engineering Resarch

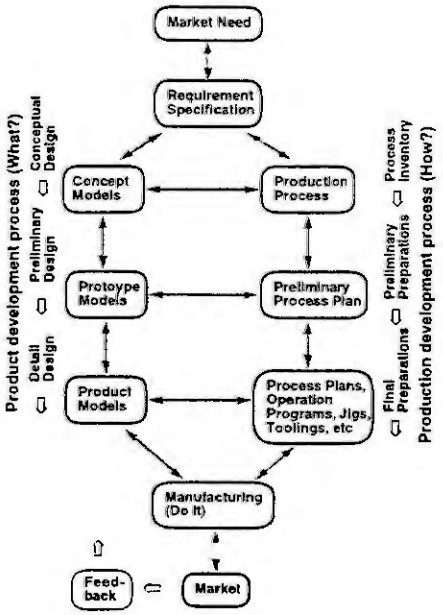


Figure 1: Parallel process development in the context of product development [1].

fields focusing on the development process of products and manufacturing systems, a driving force was established early in the project to develop methods for concurrent process development and interactive virtual collaboration.

The method related to the manufacturing systems development make use of a current product from BT Products AB as shown in figure 2a. The aim of using simulation techniques is to integrate the activities described in figure 1 and develop methods which enhance the connections between different development activities and speed up the iteration loops in the product development process with a better design of not only the product, but also the manufacturing system. Specifically, the development processes of a product will, in general, include the following parts with respect to the product:

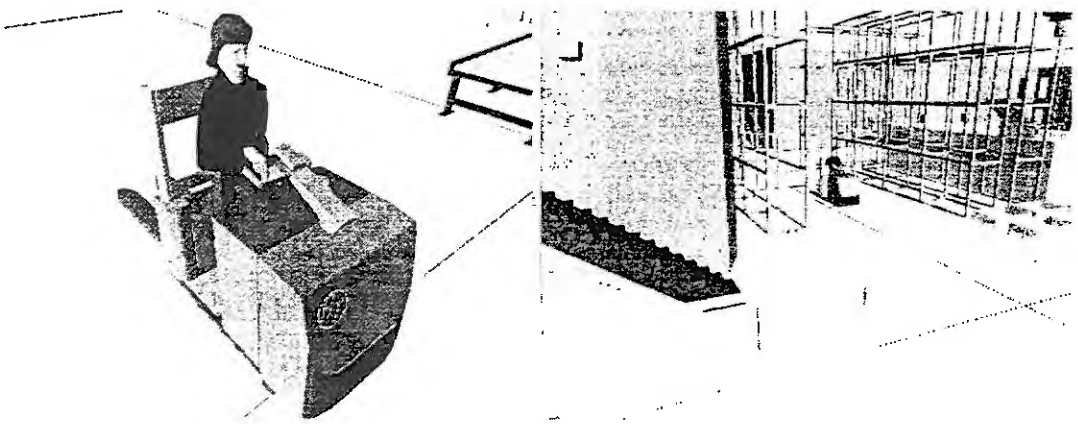


Figure 2: (a)Example of a truck which the project and simulation work use within the project to develop the methodologies, (b) Example of simulation of the truck in a shopfloor environment with a virtual user.

- **Functional simulation related to the product process development.** This activity relates to specific study of subsystems of the product including mechanical interference, motions. dynamics, packing issues, etc. Besides the subsystems a user oriented perspective will be introduced by simulating and analyzing the product in a CAVE environment.
- **Production process simulation.** This part includes specific production processes that include a limited number of operations. Examples are robotic welding station and manual assembly of the product from subsystems.

- **Production system process simulation.** This part includes the production line including subsystems and assembly of the subsystems to form the total product. Specifically, the dynamics of the production system is of importance including direct coupling in the simulation environment with both the planning tools and the PLC systems on the shopfloor.

Simulation studies focus on modeling the shopfloor where the actual production of the product take place. Moreover, this is a typical environment for the truck and represents an excellent model to test functions related to user and market needs besides simulations and studies related to the actual manufacturing processes. Figure 2b shows an example of a the truck on the shopfloor.

3. Simulation in the development process

3.1 Discrete event simulation

Within the scope of modeling the shopfloor activities for discrete event simulation a set of fundamental issues was formulated:

1. Any model that describe the activities that take part in the production line (operations, transport, etc. on the shop floor) must be able to deliver a substantial degree of precision relating to what is happening in the real world.

2. There is a trade-off between precision and resources needed to develop such models. Thus, a concurrent methodology must be developed that support incremental development as well as concurrent development by many simulation engineers a the same time.

3. Large simulation models with a high degree of precision tends to put a heavy load on computer resources. Thus, a modular approach must be used to split models in modules that enables faster development and simulation of each module as well as complete simulations over a set of net-work computers of a complete production line.

4. Model the correct routing is a complex task even for a flexible production line with a small number of machines. In almost all lines, exceptions exist that must be considered. A practical way to solve the problem and to reduce software errors is to use the code that is running in the PLC system together with other information in resource planing systems.

To address the above-mentioned issues a methodology based on incremental development was formulated. This is also in line with Pidd [7] who gives a number of principles for simulation modeling where two of them are emphasized here: (i) Start small and add, and (ii) decomposition. The methodology presented is based on those principles.

Simulation models within this context are supposed to exist in parallel with the manufacturing system over a period of time. Simulation models are often built for fast analyzes of small parts of a manufacturing system. This has its application, but we argue that simulation is more cost efficient when a holistic view of the manufacturing system is applied and where manufacturing system defects are removed continuously. Using vast resources on large models for one-shot simulations is here considered inefficient. Lead-time for a project can be measured from the start of the simulation study until the presentation of the simulation results. A more correct figure would be if the lead-time was measured until completed implementation. With this view of lead-time, fast implementation becomes an issue.

Two ways of dividing the simulation study into stages can be identified: One is to work first vertically and then horizontally, i.e. develop small simulation models of small parts of the manufacturing system and implement the results. When the sub-models are ready they are connected into one large model. The other way is to take a holistic view of the manufacturing system and generate a draft model. With this approach we work with a top-down approach, i.e. first horizontally to identify where the defects are and then vertically to identify and remove the defects.

3.2 Simulation based validation of the product

Validation of a new product includes two aspects; a user (customer/market) based study and a study how the product will be assembled from sub-assemblies in the assembly line. In our study the case was defined by the use

of the product at the producer of the truck. Specific issued to study was design and selection of steering module, control buttons, driver platform, forks (length, lift capacity).

During the simulation based validation study it was however evident that the "programming" of humans within the simulation software can not be compared with programming robots, mainly due to the complexity of the kinematics of humans. Therefore, another approach has been taken for the manual assembly simulation and the approach is to "wire" a human via sensors to the human model within the software and through this connection teach sequences within the context of a model based manual assembly operation. This work is currently ongoing and uses the sensors for motion capture of sequences.

4. Concluding Remarks

Advanced simulation techniques provide an important technology to integrate the development processes of manufacturing systems, design and market.

A method to share and distribute results over a computer network was studied and developed and has shown promising results as a valuable tool for interactive communication within the product development process.

It has been shown that a specific functionality of a product is possible to represent in a simulation environment. This has been modeled and simulated for the manual maneuvering of the truck.

A method to share and distribute results over a computer network was studied. This work was initiated due to the necessity to share information among experts at different geographical located places. This work has already shown promising results as a valuable tool for interactive communication within the product development process.

Robotic arc welding simulations showed immediate feed-back to redesign the chassis with respect to accessibility and weld joint design during welding operation. This clearly show the iterative work procedure promoted by the simulation tools.

Within the context of discrete event simulation some basic approaches have been taken to address the problem of developing large models with greater detail, where necessary. This includes incremental design, modularity and concurrent version control of the simulation model data files. The methodology supports developing large and complex models with a holistic view that provide necessary information throughout the life-cycle of a production line.

Acknowledgments

The authors are indebted to NUTEK for a grant supporting this work which is a part of the program "IT i Verkstadsindustrin", BT Products AB and Tehdasmallit AB for active and valuable support in the project work.

References

[1] B.Gustafsson. About spreading the VSOP technology. VSOP Conference III, The Swedish Institute for Production Engineering Research, 1998.

[2] O.Buckmann, M.Krömker, G.Bolmsjö, U.Lorentzon, and F.Charrier. Design of mobile robots or health care components by use of the rapid prototyping technologies. In *Proceedings of SIRS'98, 6th International Symposium on Intelligent Robotic Systems '98*, Edingburgh, July 1998.

[3] G.Bolmsjö, M.Olsson, P.Hedenborn, U.Lorentzon, F.Charrier, and H.Nasri. Modular robotics design - System integration of a robot for disabled people. In *Proceedings of EURISCON'98*, Athens, June 1998.

[4] B.Gustafsson et al. Advanced parametric programming of arc-welding robot. In *Proceedings of 28th International symposium on robotics*, Detroit, MI, October 1997.

[5] G.Bolmsjö, M.Olsson, and K.Brink. Off-line programming of GMAW robotic systems — a case study. *Int.J.Joining of Materials*, 9(3):86–92, 1997.

[6] G.Bolmsjö. Simulation of robotic assembly. In *Proceedings of the Deneb User Conference*, Troy, Detroit, MI, October 1997.

[7] M. Pidd. Five simple principles of modelling. In *Proceedings of the 1996 Winter Simulation Conference*, pages 721–728, 1996.

NEUROFUZZY HYBRID POSITION/FORCE CONTROL OF INDUSTRIAL ROBOTS: A SIMULATION STUDY FOR THE MILLING TASK.

S.G Tzafestas ,C.E Syrseloudis and G.G Rigatos.
Intelligent Robotics and Automation Laboratory
Department of Electrical and Computer Engineering
National Technical University of Athens
Zografou ,15773 ,Athens ,Greece
e-mail:tzafesta@softlab.ece.ntua.gr .

Abstract : The goal of this work is to compare neuro-fuzzy hybrid position/force control and hybrid control based on the resolved acceleration method when both of them are applied to the milling process. The convergence and stability conditions for this neuro-fuzzy learning controller are investigated. Also, an identification algorithm for the stiffness of the materials under milling is developed, and the case where the milling task is modeled by the tracking of a desirable depth curve is extensively investigated via simulation.

Introduction

Computational intelligence (CI) provides a very good set of tools for the solution of robotic tasks. CI techniques try to imitate the operation of the human brain (this is especially true for fuzzy logic and neural networks) and do not need mathematical models in order to control a system .In industrial robotics the demands for sophisticated tasks are increased. Tasks like grinding ,milling , polishing or assembling need the control of the force that the end-effector exerts on the workpieces as well as the control of its position. Hybrid position/force control, which separates the control of the robot's degrees of freedom, constitutes a good approach for handling complex tasks like the above. In this paper a neuro-fuzzy and a conventional approach to hybrid position/force control will be presented and compared.

Hybrid position/force control

The hybrid controller is based on the idea of filtering the joint torque's components which are responsible for the control of the robot's degrees of freedom (position, force), via a selection matrix S. A hybrid position/force controller (figure 1) has the following form :

$$\tau = D(q)\, J^{-1}[(I-S)u_x - \dot{J}\dot{q}] + h(q,\dot{q}) + g(q) + F_{jc} + J^T f + J^T S u_f \qquad (1)$$

where D(q) is the inertia matrix, J is the robot Jacobian, q is the vector of joint angles, h(q,dq/dt) is the vector of Coriolis and centrifugal forces, g is the vector of gravity forces, F_{jc} is the Coulomb friction of the robot joints, f is the force applied to the environment and u_x and u_f are the position and force control signals respectively.

Fuzzy neural and conventional hybrid position/force control

Consider a robotic manipulator which consists of three rotating joints and moves on a vertical plane. The controllable variables are the angle of the end-effector, the horizontal position, and the force which is applied along the vertical axis .For the fuzzy-neural controllers the range of each input signal is divided in five fuzzy sets (positive big :PB , positive small: PS, near zero :NZ , negative small: NS, negative big: NB) and n=25 rules are used at the rule layer. These fuzzy sets are kept fixed during the network operation . The membership functions of the fuzzy sets are selected to be of the Gaussian type. The fuzzy neural controller employed is depicted in figure 2.

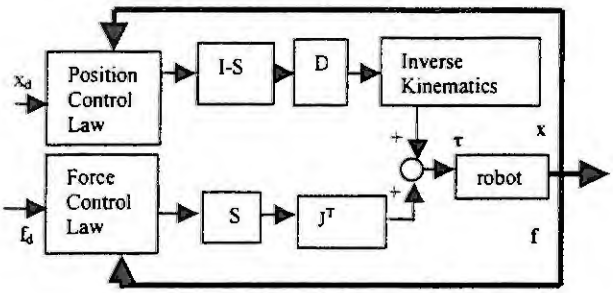


Figure 1. Hybrid position\force controller

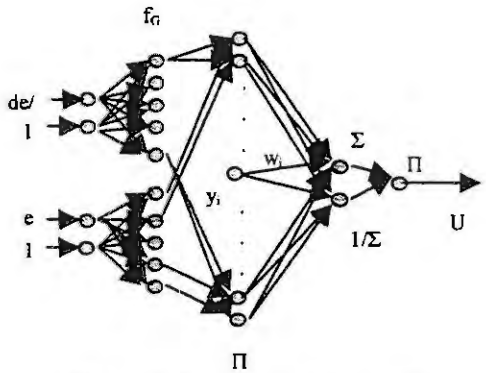


Figure 2. Fuzzy neural controller

The weight updating(learning) law for the angle control which minimizes this cost function :

$$J = \frac{1}{2} e_\varphi^2 = \frac{1}{2} [\varphi_d - \varphi]^2 \qquad (2)$$

is found to be :

$$w_i(k+1)=w_i(k) + e_\varphi (y_{i\varphi} / \sum_{i=1}^{n} y_{i\varphi}) \qquad (3)$$

where $y_{i\varphi}$ is the output from the i-th node of the rule layer and w_i is the respective weight .With the same way, the leaning rule for the horizontal motion controller and the force controller are found to be :

$$w_i(k+1)=w_i(k) + e_x (y_{ix} / \sum_{i=1}^{n} y_{ix}) \qquad (4)$$

$$w_i(k+1)=w_i(k) + e_f (y_{if} / \sum_{i=1}^{n} y_{if}) \qquad (5)$$

where n=25 is the number of rules in the controllers' rule layer .
The standard resolved acceleration controller is selected for the horizontal position and angle control ,i.e:

$$u_x = d^2x_d/dt^2 + k_{vx}(dx_d/dt - dx/dt) + k_{px}(x_d - x) \quad \text{(position control signal)} \qquad (6)$$

$$u_\varphi = d^2\varphi_d/dt^2 + k_{v\varphi}(d\varphi_d/dt - d\varphi/dt) + k_{p\varphi}(\varphi_d - \varphi) \quad \text{(angle control signal)} \qquad (7)$$

Similarly, the force controller equation is :

$$u_f = d^2f_d/dt^2 + k_{vf}\, dy/dt + k_{pf}(f_d - f) \quad \text{(force control signal)} \qquad (8)$$

Stability of the fuzzy neural networks

The interesting part of the fuzzy-neural networks, which will be studied here, is the adaptive part. This part consists of the nodes of the rule layer, the adaptable weights and the defuzzifier layer (Figure 2).
The inference engine which is adopted here is of the product-type and the defuzzifier of the center average-type .So, the nodes of the rule layer are multipliers (Π). The Σ node gives $\sum_{i=1}^{n} w_i y_i$ and the $1/\Sigma$ node gives $1/\sum_{i=1}^{n} y_i$ .The output U of the fuzzy neural network is :

$$U = \sum_{i=1}^{n} w_i y_i / \sum_{i=1}^{n} y_i = \sum_{i=1}^{n} w_i g_i \qquad (9)$$

where n is the number of rules and $g_i = y_i / \sum_{i=1}^{n} y_i$.

The matrix form of this equation is :

$$U = w^T g \quad (10)$$

where $w^T = [w_1,\ldots,w_n]$ and $g^T = [g_1,\ldots,g_n]$.

Let $U_d(k)$ be the optimal control signal which yields the desirable response .The error between this optimal control signal and the real one at the k-th instant is equal to :

$$e(k) = U_d(k) - w^T(k)g(k) \quad (11)$$

and the corresponding cost function is :

$$J(k) = \frac{1}{2}[e(k)]^2 \quad (12)$$

which by using (11) becomes :

$$J(k) = \frac{1}{2}[U_d^2(k) - 2w^T(k)p(k) + w^T(k)Y(k)w(k)] \quad (13)$$

where $p(k) = U_d(k)g(k)$ and $Y(k) = g(k)g^T(k)$ (an nxn symmetric matrix).

Differentiating (13) with respect to w(k) gives :

$$\partial J(k)/\partial w = -p(k) + Y(k)w(k) \quad (14)$$

Therefore the weight vector renewal (steepest descent) equation is :

$$w(k+1) = w(k) + \eta[-p(k) + Y(k)w(k)] \quad (15)$$

The optimal weight vector w_o is obtained by solving the equation :

$$\partial J(k)/\partial w\,|_{w=w_o} = 0 \quad (16)$$

or, by (14), the equation :

$$Y(k)\,w_o = p(k) \quad (17)$$

Introducing (17) into (15) yields :

$$w(k+1) = w(k) - \eta Y(k)[w(k) - w_o] \quad (18)$$

which can be written as :

$$c(k+1) = (I - \eta\, Y(k))\, c(k) \quad (19)$$

where $c(k) = w(k) - w_o$

The matrix Y(k) is square and symmetrical ,and so it can be written as :

$$Y(k) = Q\Lambda Q^T \quad (20)$$

where Λ is the diagonal matrix of eigenvalues and Q the eigenvector matrix of Y(k) . Using (20) along with the property $Q^T = Q^{-1}$,(19) reduces to :

$$v(k+1) = (I - \eta\,\Lambda)\, v(k) \quad (21)$$

where $v(k) = Q^T c(k)$, $v(k+1) = Q^T c(k+1)$ and $\Lambda = diag[\lambda_1,\lambda_2,\ldots,\lambda_n]$.

Component –wise, equation (21) is written as :

$$v_i(k+1) = (I - \eta\,\lambda_i)\, v_i(k) \quad (22)$$

In terms of v , the new weight vector w(k+1) is given by :

$$w(k+1) = w_o + Q\, v(k+1) \quad (23)$$

which means that the new weights $w_i(k+1)$,i=1,2,...,n are linear combinations of the $v_i(k+1)$'s .

If $\lambda_i < 0$ then $v_i(k+1) > v_i(k)$, which means that the distance from the optimal weight is increased and the convergence of the algorithm is delayed. If $\lambda_i > 0$ then $v_i(k+1) < v_i(k)$ and the distance from the optimal weight vector is decreased and the convergence is advanced .The final conclusion is that the convergence of the learning algorithm is achieved, when all the eigenvalues of the matirx Y(k) are positive .

Stiffness identification

In the milling process the knowledge of the materials' stiffness is necessary. So, stiffness identification algorithms are needed that must be performed by the manipulator to a useless experimental piece of the material ,before the actual milling process starts . This method is as follows.

Let k_n be an arbitrary initial value for the stiffness of the material to be milled, smaller than the real stiffness, and d_{max} a desirable depth .Initially a force set-point is calculated using the equation:

$$f_d = k_n\, d_{max} \quad (24)$$

The real depth d_r that has been achieved satisfies the equation :

$$f_d = k_r\, d_r \quad (25)$$

where k_r is the real stiffness of the object .

From (24) and (25) it follows that the relation of d_r and d_{max} is :

$$d_r = (k_n / k_r) d_{max} \quad (26)$$

The error e between the desirable and real depth is :

$$e = d_{max} - d_r \quad (27)$$

Now define the error cost function $J = \frac{1}{2} e^2$.

The aim of the gradient algorithm is to minimize J .Differentiating J with respect to k_n and applying the chain rule gives :

$$\partial J / \partial k_n = (\partial J / \partial e)(\partial e / \partial kn) = e\, \partial (d_{max} - k_n d_r / k_r) / \partial k_n = -e\, d_{max} / k_r \quad (28)$$

The stiffness identification algorithm cannot use k_r from (28) because it is unknown .Replacing the ratio d_{max} / k_r in (28) by its value obtained from (26) yields :

$$\partial J / \partial k_n = -e\, d_r / k_n \quad (29)$$

Therefore the updating equation of the object's stiffness is :

$$k_r(k+1) = k_r(k) + \eta\, e\, d_r / k_r \quad (30)$$

where η is the renewal rate .

Implementation of hybrid control on the milling process

In this section the implementation of the hybrid position/force control on the milling process by a desired depth curve is presented .As desired depth curve the $\sin^2$ function is selected .

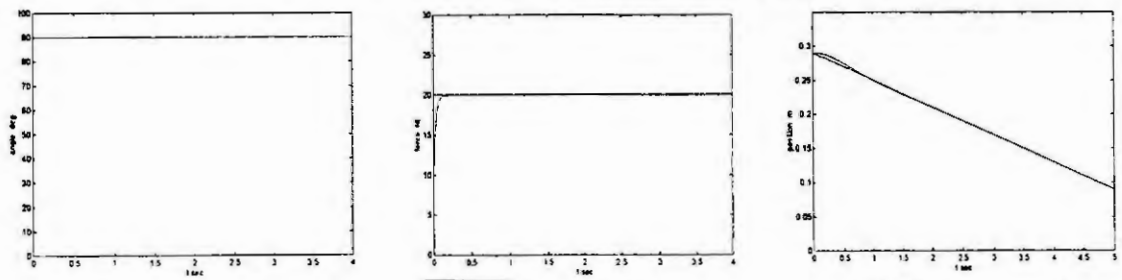
The milling process will be performed at an horizontal velocity 2cm/sec .It will be assumed that in the horizontal motion there is friction between the tool and the object which is opposite to the motion and assumed to be proportional to the real depth by a coefficient value 1500Nt/m giving a 3Nt maximum friction value at the maximum depth of 2mm.

Simulation results

The conventional resolved acceleration hybrid control method and the neuro-fuzzy hybrid control were applied using the 3-DOF robot manipulator described earlier in this paper. The robot has the following parameters : m_1 = 5kgr,m_2 =5kgr ,m_3 = 2.5kgr and l_1 = 30cm ,l_2 =30cm and l_3 =15cm respectively .The implementation was performed via computer simulation in C++ language .

The parameters of the conventional controllers were selected as: a)*Angle Controller:* $k_{p\varphi}$ = 700 ,$k_{v\varphi}$ = 1500,b)*Position Controller :* k_{px} = 15 ,k_{vx} =3.5,c)*Force Controller :* k_{pf} = 15 ,k_{vf} = 5000 .

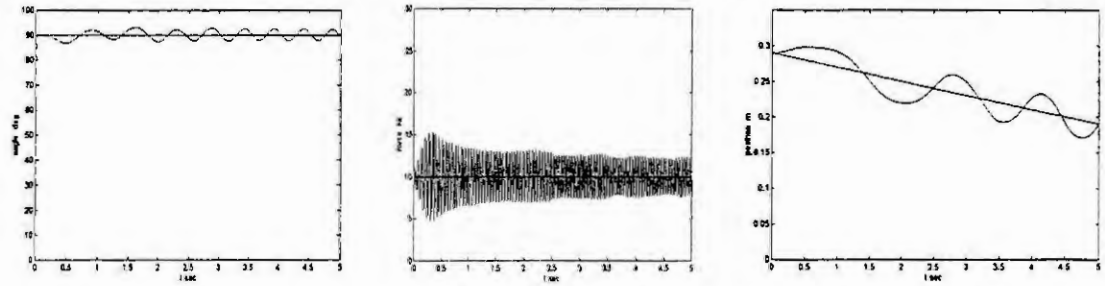
From the requirement the tool to be vertical to the object during the milling process the desired angle set–point is found to be 90° .The aim of the horizontal position controller is to move the tool with a desired velocity, and the force controller has to apply the desired force to the object . It was also assumed that there is friction between the tool and the object which is proportional to the real depth by a coefficient with value 1500 .The motion started from the point x=0.29027m and directed opposite to the x axis , while the tool was vertical to the object from the beginning . The results are shown in figure 3 :



(a)Angle control (b) Force control (c)Position control

Figure 3: Stiffness 10000Nt/m ,desired force 20Nt and desired horizontal movement by 4cm/sec.

The initialization , the direction of motion ,the control demands and the friction effects are the same as in the conventional hybrid control .In parallel with the force controller, a proportional(P) controller with gain 15 was added for reducing the oscillation effects . The simulation results are shown in figure 4, were the oscillation lines are the real values that neuro-fuzzy controllers gave and the direct lines are the desired :



(a)Angle control (b) Force control (c)Position control

Figure 4 :Stiffness 10000Nt/m ,desired force 10Nt and desired horizontal movement by 2cm/sec.

The effectiveness of the stiffness identification algorithm was evaluated by using several materials with different stiffness values .The conventional resolved acceleration controllers were employed with the tool vertical to the object and without horizontal motion . The simulation results obtained are shown in figures 5 and 6 :

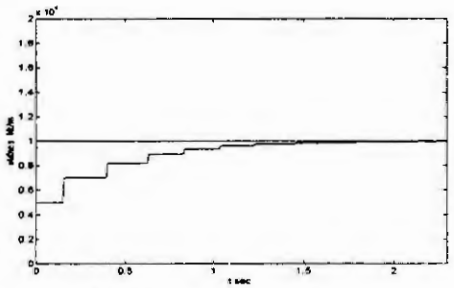


Figure 5:Real stiffness 10000Nt/m

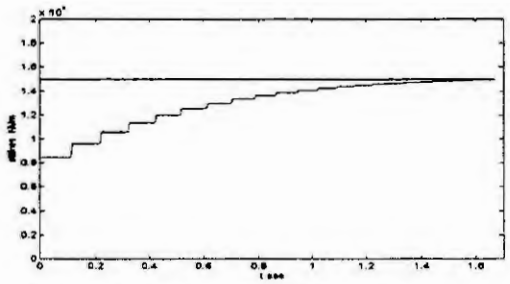


Figure 6: Real stiffness 15000Nt/m

After the stiffness' identification, the milling process was performed using the conventional resolved acceleration hybrid control method .The controllers' parameters were the same as before except from k_{pf} which was changed to k_{pf} =800 for better tracking . The process was started from the position x=0.29027m with the tool vertical to the object and finished at the point x=0.19027m. The simulation results are shown in figures 7 and 8 :

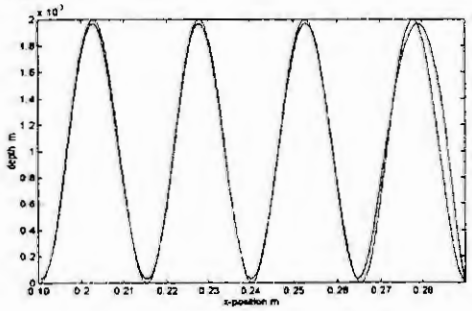


Figure 7: Stiffness 10000Nt/m and desired horizontal movement by 2cm/sec.

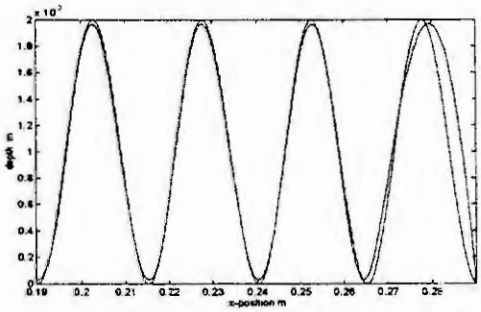


Figure 8: Stiffness 15000Nt/m and desired horizontal movement by 2cm/sec.

Conclusions

As demonstrated by the simulation results ,the fuzzy neural controller did not show a satisfactory behavior while the conventional resolved acceleration controller showed a good performance .The fuzzy neural hybrid position/force control has shown oscillation behavior, coming obviously from the inherent stability problems of the gradient algorithm . The matrix Y of the fuzzy neural networks used as controller was a 25x25 dimensional matrix . For this reason the probability to have negative eigenvalues and so to present oscillations was very high .
A stiffness identification algorithm was also developed that was embodied to the conventional resolved acceleration controller which had a better performance. Two kinds of materials were subject to milling and a desirable depth curve was followed .

References

[1] Fukuda.T –Kiguchi.K ,Intelligent Position/Force Control for Industrial Robot Manipulators : Application of Fuzzy-Neural Network , *IEEE Trans.Ind.Electronics Vol. 44 , No 6 ,December 1997.*

[2] Haykin Simon ,Adaptive Filter Theory (2^{nd} Ed.) , *Prentice Hall, Information and Systems Sciences Series , 1991 .*

[3] Kazerooni H. - Sheridan T.B - Houpt P.K. ,Robust Compliance Motion for Manipulators : Part I-The Fundamental Concepts of Compliant Motion , "Part II-Design Method ", *IEEE Trans.Robot.Automat. , Vol. RA-2, No 2, pp. 83-105, 1986.*

[4] Kiguchi K. - Fukuda T. Fuzzy Neural Controller for Robot Manipulator Force Control , *Proc. Joint Conf. 4^{th} IEEE Int. Conf. Fuzzy Systems and 2^{nd} Ind. Fuzzy Engineering Symp. , Vol. 2, pp. 869-874, 1995.*

[5] Kiguchi K. - Fukuda T., Robot Manipulator Contact Force Control Application of Fuzzy-Neural Networks , *Proc. IEEE Int. Conf. Robotics and Automation , Vol. 1, pp. 875-880, 1995 .*

[6] Luh J.Y.S. - Walker W. M. - Paul R. P. C., Resolved Acceleration Control of Manipulators , *IEEE Trans. Automat. Contr. , Vol. AC-25, pp.468-474, 1980 .*

[7] Raibert M. H. - Craig J. J. , Hybrid Position/Force Control of Manipulators , *ASME J. Dyn. Syst. Meas. Control , Vol. 102 , pp.126-133 June 1981.*

[8] Wang Li-Xin ,Adaptive Fuzzy Systems and Control : Design and Stability Analysis , *Prentice Hall , Englewood Cliffs ,1994 .*

Modelling and Simulation of Unsteady Heat Transfer Effects on Trajectory Optimization of Aerospace Vehicles

M. Dinkelmann[1], M. Wächter[2] and G. Sachs[2]
[1]Daimler-Chrysler Aerospace, Military Aircraft Division, D-81663 Ottobrunn
[2]Institute of Flight Mechanics and Flight Control, Technische Universität München
Boltzmannstr. 15, D-85748 Garching

Abstract. Heat input reduction by optimal trajectory control is considered for the range cruise of a hypersonic flight system propelled by a turbo/ram jet engines combination. A mathematical model is developed for describing the unsteady heat transfer through a thermal protection system. This model is coupled to the equations of motion of the vehicle. An efficient optimization technique is applied for constructing a solution. The results show that a significant heat input reduction can be achieved with only a small increase in fuel consumption.

Introduction

The hypersonic flight regime has received a great interest in recent years because new concepts of aerospace plane type vehicles are considered as a means for reducing the costs and providing an improved space transport capability. For some of these concepts, an aerodynamic lifting capability and airbreathing propulsion is a common feature.

Hypersonic flight poses challenging problems. One of the major concerns is the hot environment to which the vehicle is exposed. This leads to the requirement of sophisticated thermal protection systems to withstand the high temperatures at hypersonic speed [7]. Such a system should avoid too hot temperatures in the inner parts of the vehicle and reduce the heat input. This problem is considered from a flight control standpoint and it will be shown that trajectory optimization is a means to reduce the heat input into the vehicle. Particular emphasis is placed on the unsteady effects in the heat transfer from the hot airflow outside of the vehicle to its interior. Because of extreme performance requirements on hypersonic vehicles, fuel minimization is a primary goal in trajectory optimization.

There is significant research in trajectory optimization of hypersonic vehicles (e.g., Refs [1] - [5]) and important results and significant progresses have been achieved, including papers on heat loads and heat input [10].

The purpose of this paper is to consider and optimize the trajectory of an airbreathing aerospace plane, with particular emphasis put on realistically modelling the unsteady heat transfer effects. It will be shown that significant reductions of the heat input can be achieved by an appropriate control of the trajectory.

The technique proposed in this paper for reducing the heat input is applied to the range cruise of a hypersonic vehicle.

Heat Input Modelling

A mathematical model is developed for describing the heat input from the hot outside airflow through the thermal protection system into the vehicle [6]. The type of vehicle considered is a hypersonic flight system with an airbreathing powerplant (turbo/ram jet engines combination).

The wall consists of various elements including layers for thermal insulation and protection (Fig. 1). To describe the heat flux an one-dimensional thermal knot model is applied (Fig. 2). The heat flux q into the wall can be expressed as

$$\begin{aligned} q_1 &= q_{air} - \varepsilon\sigma\left(T_1^4 - T_\infty^4\right), \\ q_i &= \left(T_{i-1} - T_i\right)\left(\frac{\lambda}{d}\right)_{i-1,i} + \left(T_{i-1}^4 - T_i^4\right)\varepsilon_{i-1,i}\,\sigma, \qquad i = 2, \cdots, n. \end{aligned} \tag{1}$$

The differential equations for the temperatures of the first $n-1$ knots read

$$\dot{T}_i = \frac{q_i - q_{i+1}}{\rho_i c_i d_i}. \tag{2}$$

For the most inner knot, there may be temperature changes yielding (Fig. 1, left side)

$$\dot{T}_n = \frac{q_n - \alpha_q \left(T_n - T_{\text{inside}}\right) - \varepsilon \sigma \left(T_n^4 - T_{\text{inside}}^4\right)}{\rho_n c_n d_n} . \tag{3}$$

For the tank wall which is cooled by liquid hydrogen the following assumption is applied

$$\dot{T}_n = 0 . \tag{4}$$

The number of knots used for the problem considered in this paper is $n = 13$.

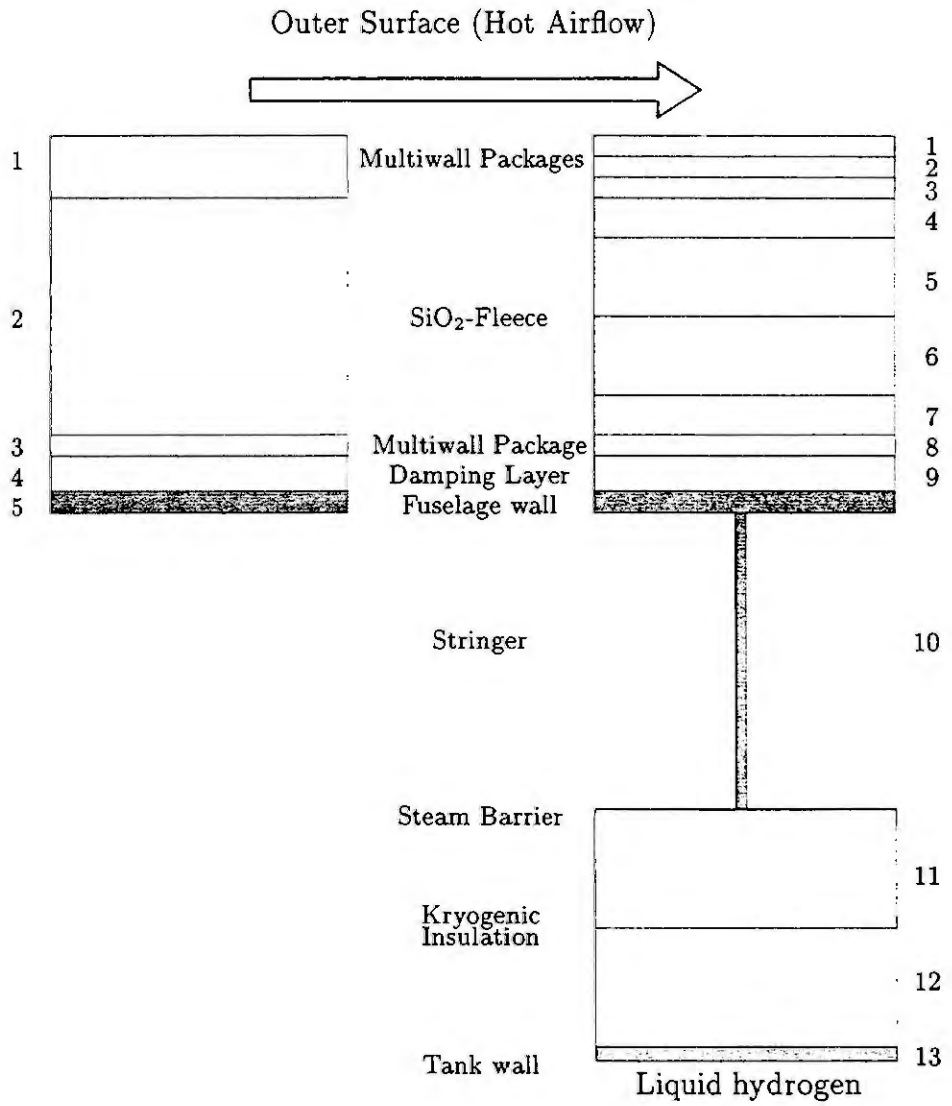


Figure 1: Wall structure with heat protection (left side: 5-layer-model without tank insulation; right side: 13-layer-model with tank insulation)

The convective heat flux q_{air}, for which a realistic and complex model is applied, depends on the outside temperature T_1, Mach number M, altitude h and angle of attack α, $q_{\text{air}} = q_{\text{air}}(T_1, M, H, \alpha)$. Thus, it is not constant but changes with the flight condition, i.e., with M, h, α. Furthermore, T_1 also depends on the flight condition, $T_1 = T_1(M, H, \alpha)$.

Before take-off the temperature in the walls is equal to the temperature of the nitrogen which runs through the structure to avoid icing during filling up the hydrogen tank.

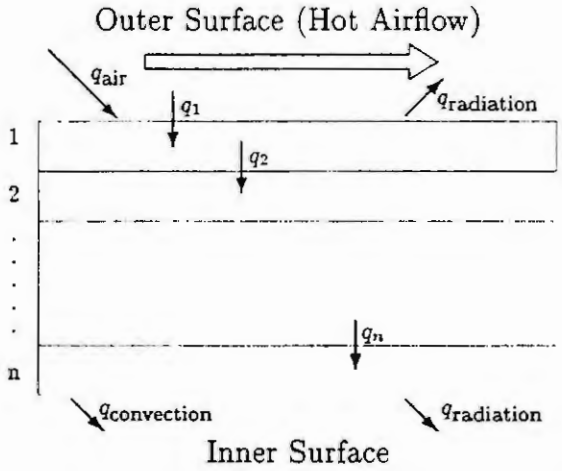


Figure 2: Heat flux model with thermal knots

Vehicle Modelling and Optimal Control Problem

The modelling of the vehicle for describing its motion is based on point mass dynamics [9]. The following relations for a range cruise apply:

$$\begin{aligned} \dot{V} &= \frac{1}{m}(F\cos\alpha - W) - g\sin\gamma + \omega_E^2 r\,\sin\gamma\,, \\ \dot{\gamma} &= \frac{1}{mV}(F\sin\alpha + A) + \cos\gamma\left(\frac{V}{r} - \frac{g}{V} + \frac{\omega_E^2 r}{V}\right) + 2\,\omega_E\,, \\ \dot{h} &= V\sin\gamma\,, \\ \dot{m} &= -\dot{m}_B\,. \end{aligned} \tag{5}$$

To describe the aerodynamics and powerplant characteristics a complex mathematical model involving multifunctional dependencies is applied.

The optimal control problem is to minimize the consumed fuel for a range cruise over 9000 km. Therefore the cost functional can be written as

$$\Phi\,(x(t_f), u(t_f)) = -m|_{t_f} \longrightarrow Min \tag{6}$$

The state variables are speed V, flight path angle γ, altitude h and vehicle mass m. The control variables are angle of attack α and throttle setting δ. State constraints for the load factor and the dynamic pressure as well as control constraints are applied for realistically describing the optimal control problem.

Furthermore, the heat input at the most interior wall layer is treated as a constraint:

$$\int_0^{t_f} q_n\,dt \leq q_{n_{limit}}\,. \tag{7}$$

This relation yields a coupling of the differential equation systems of the unsteady heat input with those of the dynamics of the vehicle, Eqs. (2) – (4) with Eq. (5). To solve the resulting boundary problem an efficient and well tested optimization technique is used [8].

Results

Results are presented in Figs. 3 and 4 where an optimal range cruise trajectory with no heat input constraint is used as a reference.

Heat input characteristics at the lower side of the fuselage under the liquid hydrogen tanks are illustrated in Fig. 4, showing the time histories of the temperatures of each layer of the wall and the heat input. While the temperatures of the outer layers are in consonance with the momentary flight condition, the inner layers show delayed changes. This particular concerns the time where the maximum

temperature is reached. Although the temperatures of the outer layers show a significant decrease in the final part of the cruise the inner layers still stay at a high level. This results from the unsteadiness of the heat flux through the wall. It may be pointed out that a steady-state model for computing

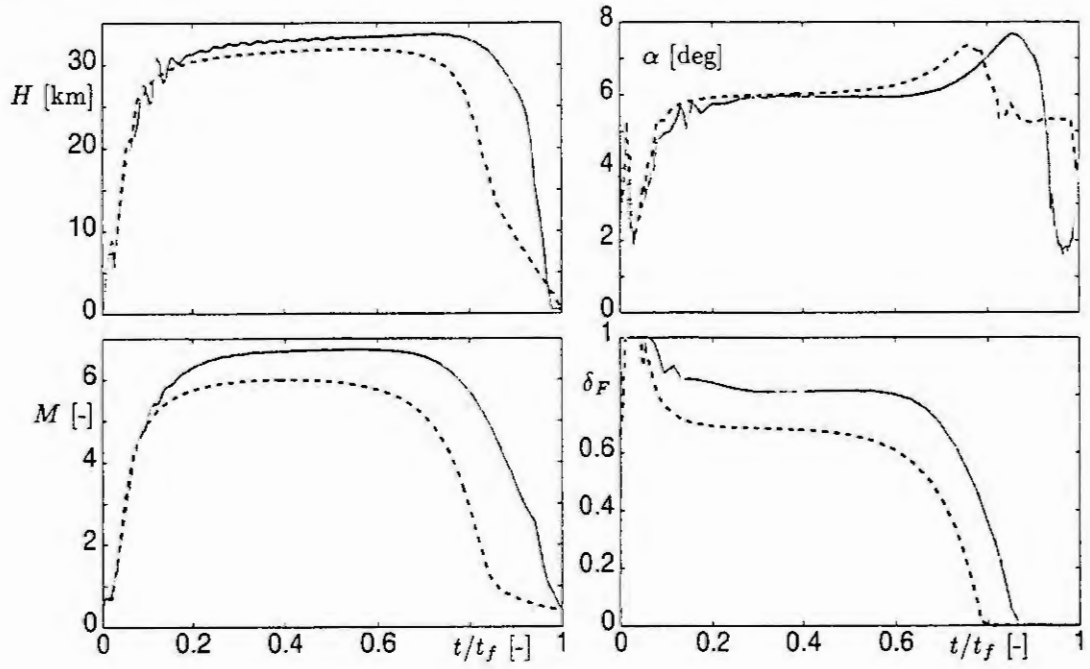


Figure 3: State and control variables of optimal 9000 km range cruise
- - - - - no heat input constraint (q_n = 1214 kJ/m^2, m_B = 62654 kg, t_f =113.3 min)
——— heat input constraint (q_n = 950 kJ/m^2, m_B = 64459 kg, t_f =89.82 min)

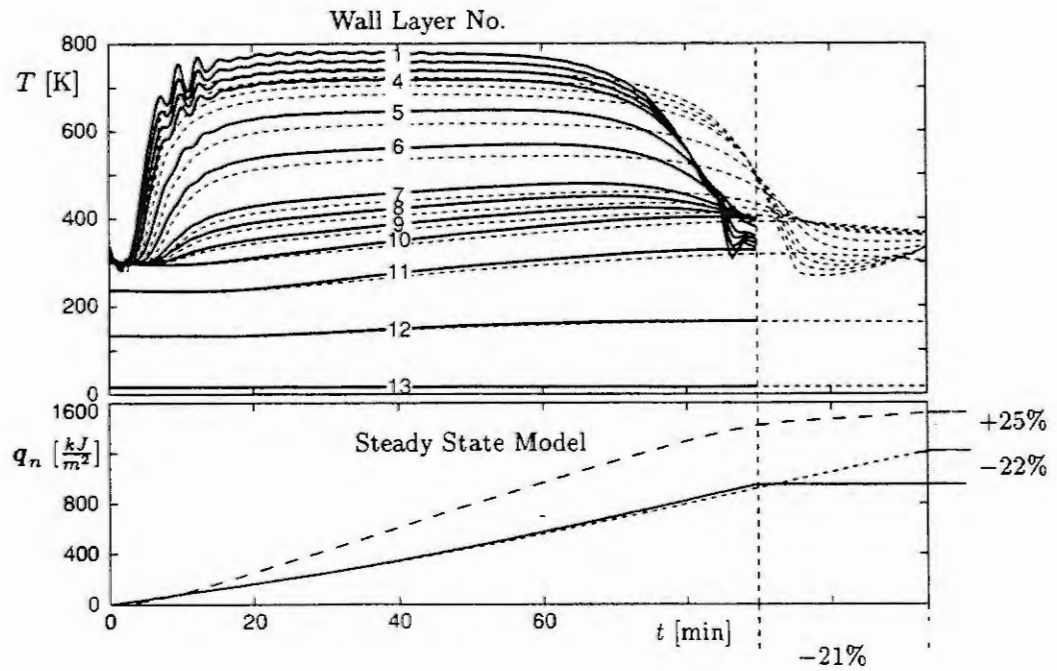


Figure 4: Temperature in the wall layers at the lower side and heat input through the most inner layer
- - - - - no heat input constraint (q_n = 1214 kJ/m^2, m_B = 62654 kg, t_f =113.3 min)
——— heat input constraint (q_n = 950 kJ/m^2, m_B = 64459 kg, t_f =89.82 min)

the temperatures and the heat flux yields a 25 % higher value for the heat input into the most inner layer (Fig. 4).

By introducing the constraint Eq. 7 the heat input can be reduced. The temperature time histories show significant differences (Fig. 4). The maximum temperature at the outer layer is increased when compared with the unconstrained reference case. The inner layers show only a slightly higher temperature level. Accordingly, the heat flux into the most inner layer would be similar. But the overall heat input is reduced due to decrease of the flight duration. As a result, a significant reduction of the heat input can be achieved, while the associated increase in fuel consumption is comparatively small.

Conclusions

The reduction of heat input by appropiate control of the flight path for a range cruise of a hypersonic flight system is considered which is equipped with a turbo/ram jet engines combination.

A complex mathematical model is developed for describing the heat flux through the thermal protection system, with emphasis placed on unsteady heat transfer effects. The differential equations for describing the heat input are coupled to the equations of motion for which a complex aerothermodynamics and powerplant modelling is used.

Solutions for the examined problem are constructed using an efficient optimization technique. The results show a significant heat input reduction, while there is only a small increase in fuel consumption.

References

[1] R. Bayer and G. Sachs. Optimal Return-to-Base Cruise of Hypersonic Carrier Vehicles. *Zeitschrift für Flugwissenschaften und Weltraumforschung*, 19(1):47–54, 1995.

[2] W. Buhl, K. Ebert, and H. Herbst. Optimal Ascent Trajectories for Advanced Launch Vehicles. In *AIAA 4th International Aerospace Planes Conference, Orlando, FL, AIAA-92-5008*, Decemher 1992.

[3] R. Bulirsch and K. Chudej. Combined Optimization of Trajectory and Stage Separation of a Hypersonic Two-Stage Space Vehicle. *Zeitschrift für Flugwissenschaften und Weltraumforschung*, 19(1):55–60, 1995.

[4] E. Cliff, K. Schnepper, and K. Well. Performance Analysis of a Transatmospheric Vehicle. In *AIAA 2nd International Aerospace Planes Conference, Orlando, FL, AIAA-90-5257*, October 1990.

[5] J. E. Corban, A. J. Calise, and G. A. Flandro. Rapid Near-Optimal Aerospace Plane Trajectory Generation and Guidance. *Journal of Guidance, Control, and Dynamics*, 14(6):1181–1190, 1991.

[6] M. Dinkelmann. *Reduzierung der thermischen Belastung eines Hyperschallflugzeugs durch optimale Bahnsteuerung.* PhD thesis, Lehrstuhl für Flugmechanik und Flugregelung, Technische Universität München, May 1997.

[7] H. Grallert and K. Vollmer. Conceptual Design of the Reference Concept SÄNGER by Means of Advanced Methods. In *AIAA/DGLR 5th International Aerospace Planes and Hypersonics Technologies Conference, München, AIAA-93-5085*, December 1993.

[8] N. N. *GESOP (Graphical Environment for Simulation and Optimization), Softwaresystem für die Bahnoptimierung.* Institut für Robotik und Systemdynamik, DLR, Oberpfaffenhofen, 1993.

[9] N. X. Vinh, A. Busemann, and R. D. Culp. *Hypersonic and Planetary Entry Flight Mechanics.* Ann Arbor, The University of Michigan Press, 1980.

[10] R. Windhorst, M. D. Ardema, and J. V. Bowles. Minimum Heating Reentry Trajectories for Advanced Hypersonic Launch Vehicles. In *AIAA Guidance, Navigation, and Control Conference, New Orleans, LA, AIAA-97-3535*, August 1997.

ADAPTATION OF THE BALANCED REALIZATION TO THE COUPLING OF REDUCED ORDER MODELS FOR THE MODELLING OF THE THERMAL BEHAVIOR OF BUILDINGS

C. Ménézo*, H. Bouia, J. J. Roux and J. Virgone
Centre de Thermique de Lyon – Equipe Thermique du Bâtiment
I.N..S.A. of Lyon - Bât. 307 - 20, Av. A. Einstein
69621 Villeurbanne Cedex – France
*corresponding author : tel. 33.4.72.43.84.59, fax. 33.4.72.43.85.22,
e-mail : christophe.menezo@insa-cethil-etb.insa-lyon.fr

Abstract. This work is devoted to the field of building physics and related to the reduction of heat conduction transfer models. The aim is to enlarge the model libraries of heat and mass transfer codes through limiting the considerable dimensions reached by the numerical systems during the modelling process of a multizone building. We show that the balanced realization technique, specifically adapted to the coupling of reduced order models with the other thermal phenomena, turns out to be very efficient.

Introduction

Buildings are thermal systems with complex design and extended area. When modelling, the scales are depending upon either the interest in the energetic behavior of the whole building or in a particular dwelling zone. This system is submitted to numerous internal and external thermal impact factors that modify its state according to a large scale of times. The different mechanisms of heat transfers are clearly identified and an accurate modelling of each of them is mastered : heat conduction through the envelope, radiation, convection and heat transfers between zones connected with mass transfer. The main problem lies in the fact that these phenomena may vary according to an interrelated manner and, very often with the same impact. The modelling of all these interconnected phenomena still remains difficult to achieve: a reasonable balance between the expected accuracy of results and the computation times has to be maintained.

We present in this paper a reduction technique applied to conductive systems. The adopted way is based on a sub-structuring approach leading to an automatic generation of reduced order models, associated with elementary components of the envelope : 1D (multi-layer walls), 2D (complex walls, thermal bridges) or 3D (bonding building-ground). This work has been aimed towards performing (in transient rate) and practical (interface CAD) codes allowing Consulting Engineers to conceive and rapidly analyze the airflow-thermal behavior of buildings.

Conduction models

In buildings, three different heat transfer mechanisms evolve in a interconnected manner and, usually, with a similar importance: conduction, radiation, convection and mass transfer. The equations ruling the thermal evolution of this structure reflect the coupling of these phenomena through the energy conservation.

The conduction evolving within the envelope can only be reasonably described by laws which are both linear and time invariant. Mathematical developments allowed to express accurately these diffusion phenomena. However, the building envelope is a complex geometric domain made of many peculiarities. The thermal transfers are in fact at least 2D. Resorting to numerical methods implies a spatial discretization of the heat equation and leads to a set of differential equations (1.a). They allowed to relate the evolution of the temperature field $T(M_i,t)$ of the n control volumes to the time varying p excitations gathered in vector $U(t)$. An observation equation (1.b) is associated with this system and enables to follow the evolution of a number q of observed variables set in the vector $Y(t)$.

$$\begin{cases} Capa.\dot{T}(M_i,t) = A_0.T(M_i,t) + B_0.U(t) & (1.a) \\ Y(t) = C_0.T(M_i,t) + D.U(t) & (1.b) \end{cases}$$

Capa is the matrix of the thermal capacities associated with each elementary volume i. This matrix is diagonal (Finite Differences or Finite Volumes) and is defined as positive. Matrix A_0 is made of the thermal conductivities linking the elementary volumes. This matrix has negative and dominant terms on its diagonal and is also symmetric. B_0 is the matrix of the driving forces, C_0 is the observation matrix and D the direct transmission matrix.

Besides, it is important to highlight that this type of model is built in order to carry out afterwards the coupling with the other thermal phenomena. For this, the p-outputs (fluxes led on the surfaces) are observed on the boundaries of each envelope component where the q-exitations are applied too (surface temperatures) and with : $p = q$. The excitation matrix B_0 and the matrix of observation C_0 are thus made with the same thermal conductivities and are equal in transposed : $B_0^T = C_0$. This is very interesting for the use of the reduction technique called balanced realization suggested by Moore [1].

Adapted balanced realization technique

The dimension of system (1), which depends on the mesh accuracy, can increase considerably for a multizone building. So, it is necessary to reduce the set of differential equations.

The methods related to the reduction of invariant linear systems consist in selecting among the n state variables of the system, the r variables considered as preponderant for the model evolution. The formulation of the original detailed model (1) and the selection criteria differ with the method. The aim is to achieve a reduced model of order $r << n$ reproducing the most accurate dynamic behavior of the detailed model.

We present first the chosen initial formulation before dwelling on Moore's technique: the symmetric or balanced modal form. This formulation will enable (see *infra*) to reduce quite considerably the numerical effort in comparison with the algorithm used on *a priori* any systems.

Thus, the state equation (1.a) is expressed as follows: $C_{apa}^{1/2} . C_{apa}^{1/2} \dot{T} = A_0 T + B_0 U$.

The change of variables, $\theta = C_{apa}^{1/2} . T$, is then applied. The system (1) is expressed as follows:

$$\begin{cases} \dot{\theta}(t) = A.\theta(t) + B.U(t) \\ Y(t) = C.\theta(t) + D.U(t) \end{cases} \text{ where : } A = C_{apa}^{-1/2} . A_0 . C_{apa}^{-1/2} , B = C_{apa}^{-1/2} . B_0 , C = C_0 . C_{apa}^{-1/2} \quad (2)$$

It is noticeable that the equality in transposed between the excitation and the observation matrices is still checked. This balance would have been broken by multiplying on the left each element of the equation (1.a) by the matrix C_{apa}^{-1} as is usually done. Besides, the state matrix A is symmetric contrarily to the matrix $C_{apa}^{-1} . A_0$ and has the same eigenvalues (they are real and strictly negative). Its diagonalization leads to the modal form (3).

$$\begin{cases} \dot{X}_t = W.X_t + \Gamma.U_t \\ Y_t = H.X_t + D.U_t \end{cases} \quad (3)$$

The differential equations are then uncoupled and thus can be solved separately. A eigenvalue λ_i is associated with each eigenvector set in P such as $W = P^{-1}.A.P = diag\ (\lambda_i)$. The equation (2), based on the properties of the thermal conduction systems leads to the basic following relation concerning the matrix of changing base : $P^{-1} = P^T$.

This symmetric modal form reduce the calculation times linked to the reversal of matrix P and allows to get rid of numerical problems linked to the reversal of big sizes matrices. In this base the control matrix Γ and the observation matrix H are expressed as follows:

$$\Gamma = P^T . C_{apa}^{-1/2} . B_0 \text{ et } H = C_0 . C_{apa}^{-1/2} . P \quad (4)$$

This enables to keep, the relation of equality between the control and observation matrices : $\Gamma^T = H$.

The general algorithm of Moore allows to preserve among the n variables of the model, the r state components which are considered as preponderant for the evolution of the model. Their evolution is deeply influenced by the excitations (they are the most controllable) and the transmission of their effects to the observed variables is important (the most observable). The gramians of controllability Wc and of observability Wo are two matrices which quantify these two notions and are related to the state variables of systems such as (2) and (3). They are solutions to Lyapunov's equations (5) and are defined as positive.

$$A.Wc + Wc.A^T = -B.B^T \text{ and } A^T.Wo + Wo.A = -C^T.C \quad (5)$$

The matrix Wc evaluates the influence of the control U on each state variable. As for the matrix Wo, the influence of each state variable on the observation Y can be defined.

Moore's method consists in looking for a particular formulation of the original system such as the gramians are equal and diagonal: $Wc = Wo = \Sigma = diag(\sigma_1, \ldots \sigma_n)$. In this new base, the state variables are thus as much controllable as observable. The matrix Π enabling to achieve this balanced base is called balancing transformed.

A robust algorithm of the determination of Π has been developed in the field of automatic and control [1]: determination of the gramians Wc and Wo – Choleski's decomposition of one of them $Wo = R^T.R$ - unitary diagonalisation of the matrix $R^T.Wc.R$ into $V^T.\Sigma^2.V$ with $V^T.V = I$ and $\Sigma = diag(\sigma_1, \ldots, \sigma_n)$ such as $\sigma_1 \geq \ldots \geq \ldots \sigma_n$ - determination of the transformed matrix $\Pi = \Sigma^{-1/2}.V^T.R$.

The number of stages is relatively important when the system dealt with is *a priori* without peculiar property concerning control and observation. Moreover, numerical problems can be seen if some variables are very weakly controllable and weakly observable (matrix Π can be numerically singular) [2]. Otherwise, when we directly use as initial form, the symmetric modal form (3), the determination of the balancing transformed is immediate. Indeed, the control and observation matrices Γ and H being equal in transposed, gramians are equal : $Wc = Wo$.

An unitary diagonalization of one of the two gramians is thus necessary to achieve the balanced realization such as defined by Moore: $Wc = \Pi^T . \Sigma . \Pi$, with $\Pi^T . \Pi = I$

The expression of the balanced estimator is thus easily achieved. The formulation of the system (3) in the new base ($\xi = \Pi . X$) is :

$$\begin{Bmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \end{Bmatrix} = \left[\begin{array}{c|c} Ae_{11} & Ae_{12} \\ \hline Ae_{21} & Ae_{22} \end{array}\right] \cdot \begin{Bmatrix} \xi_1 \\ \xi_2 \end{Bmatrix} + \begin{bmatrix} Be_1 \\ Be_2 \end{bmatrix} \cdot U \quad \text{and} \quad Y = \left[\begin{array}{c|c} Ce_1 & Ce_2 \end{array}\right] \cdot \begin{Bmatrix} \xi_1 \\ \xi_2 \end{Bmatrix} + [D] \cdot U \tag{6}$$

The r prevailing state variables are contained in the vector ξ_1. They are identified from the r biggest singular values $\Sigma_1 = diag(\sigma_1, \ldots, \sigma_r)$. r is determined in a recurrent way: a limit is established (99 % of the total sum of singular values). Beyond this limit, the participation of the state variables (which are associated with $\Sigma_2 = diag(\sigma_{r+1}, \ldots, \sigma_n)$) to the transmission of the effect of excitations U towards observation variables Y, is considered as negligible.

Elimination of the $(n\text{-}r)$ weakest controllable and observable variables is then carried out by truncation [3], preserving the static rate between the detailed model (6) and the reduced one (7).

$$\begin{cases} \dot{\xi}_1 = \left(Ae_{11} - Ae_{12} . Ae_{22}^{-1} . Ae_{21}\right) \cdot \xi_1 + \left(Be_1 - Ae_{12} . Ae_{22}^{-1} . Be_2\right) \cdot U \\ \tilde{Y}(t) = \left(Ce_1 - Ce_2 . Ae_{22}^{-1} . Ae_{21}\right) \cdot \xi_1 + \left(D - Ce_2 . Ae_{22}^{-1} . Be_2\right) \cdot U \end{cases} \tag{7}$$

Application to the modelling of a dwelling cell

In order to illustrate the efficient behavior of the reduced models, the modelling of a monozone cell is realised. This experimental cell, CIRCE (figure 1) belongs to the CoSTIC (Technique and Science Committee of Climatic Industries). A detailed description of the cell as well as the used metrology are to be found in [4]. The north wall is equipped with a simple glazing. Each wall is in contact with a climatic caisson which regulate the outside ambiences. A system of air supply and a heating floor (figure 2) are available. The scenarios of the tests have been carried out on a period of 10 days. They have been designed in order to get a dynamic rate by modifying by steps : the climatic caisson temperatures, the air supply conditions and on the temperature of the coolant fluid.

The walls have been discretized in 1D, except for the floor (2D). Its decomposition into 68 elementary frame (figure 2) is defined from the symmetries of the spatial distribution of the draining network. The reduced models associated with the envelope components (table I) are coupled to each other by non-linear internal radiation heat transfers and to the air volume by convection heat transfers. This numerical coupling technique, of *ping-pong* type, enables to take into account non-linear exchange laws for convective and radiation heat transfers (*see* [3]).

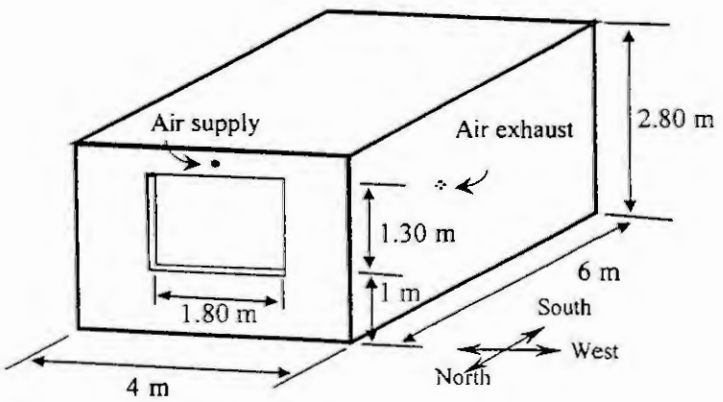


Figure 1 *Dwellings cell*

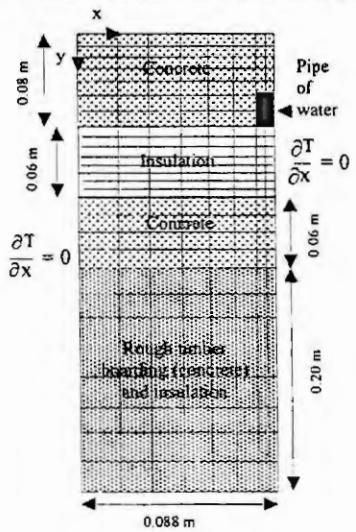


Figure 2 *Elementary frame of the heating floor*

The iterative calculation is first, simultaneously carried out from the balances on the air volume and the inner faces of the envelope components. Below a given convergence limit, the resolution time step of the conductive systems (7) is incremented.

Model	North Wall	Glazing	West Wall	East Wall	South Wall	Heating floor	Ceiling
Detailed: n	10	1	10	10	10	196	10
Reduced : r	2	1	2	2	2	7	2

Table I *Detailed and reduced orders of the enveloppe components*

The simulations presented on figures 3 and 4 have been carried out from 2 non-linear laws describing convective exchanges at the inner surface of the heating floor. They result from experimental studies on that kind of heat source and are found in literature. The results are listed in figure 3 for the air temperature and figure 4 for the surface temperature of the floor. The uncoupling of the model responses appears after the first step imposed on the coolant fluid temperature. Considering the observed scattering on the correlations [3], these gaps are not ascribable to the reduced conductive models. An intermediate empirical correlation, enabling us to improve noticeably the results, is possible. It is noticeable that the experimental conditions are extreme. In practice, the surface temperature of the floor is limited to 28 °C for comfort reasons. This would normally lead to a reduction of the gaps noticed between simulations and the measures.

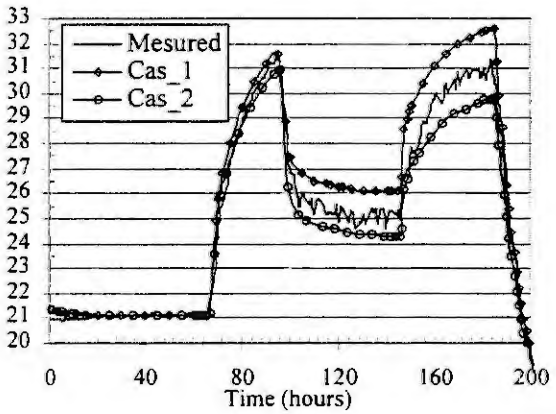


Figure 3 *Inside Mean Ambient Temperature (°C)*

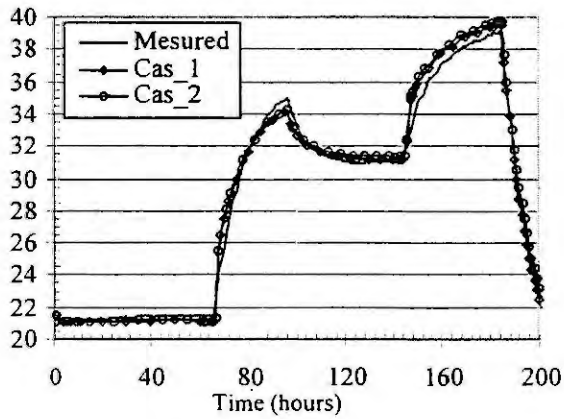


Figure 4 *Floor Inner side Mean Temperature (°C)*

Conclusions

The algorithm suggested by Moore has originally been developed to deal efficiently with linear systems of any shape. The algorithm, suggested here and built on subsequent coupling considerations, limits the steps of the reduction technique. Indeed, obtaining the gramians equality, even before any treatment, enables to limit the numerical handling for the determination of the balancing transformation. The reduced models produced by this method, have a very stable dynamical behavior (mainly at the level of transient rates) experimentally confirmed. We are now working on the coupling of reduced models with more accurate mass transfer models such as zonal models [5]. The first results are encouraging, in term of accuracy of results and numerical effort to provide.

References

1. C. Moore, Principal Component Analysis in Linear Systems: Controllability, Observability, and Model Reduction. IEEE Trans. on Contr., Vol. AC-26, n° 1 (1981), 17-32.
2. G. Safonov, R.Y. Chianga, Schur Method for Balanced-Truncation Model Reduction. IEEE Trans. on Contr., Vol. 34, n° 7 (1989), 729-733
3. C. Ménézo, Contribution à la modélisation du comportement thermique des bâtiments par couplage de modèles réduits, PhD Thesis, INSA de Lyon, 1999, 260 p.
4. D. Palenzuela, GREC : Etalonnage de la cellule d'essais CIRCE, CoSTIC, Digne, 1993, 95 p.
5. C. Inard, H. Bouia, P. Dalicieux, Prediction of air temperature distribution in buildings with a zonal model. Energy and Buildings, vol 24, n° 2 (1996), 125-132.

MODELING THE HEATING OF A BUILDING SPACE USING MATLAB-SIMULINK

M M Gouda BSc MSc *, **S Danaher** PhD CPhys CEng MIEE *, **C P Underwood** PhD CEng MCIBSE MASHRAE §
* School of Engineering, University of Northumbria, Newcastle upon Tyne NE1 8ST, UK
§ School of the Built Environment, University of Northumbria, Newcastle upon Tyne NE1 8ST, UK

1- ABSTRACT

Significant progress has been made in recent years on the development of modular and generic simulation programs for investigating the thermal behaviour of buildings and associated HVAC plant and controls. Many of these programs are however inflexible for the specific analysis of HVAC plant and control systems over short timescales. The building space model is expressed as a linear time-invariant state-space description. A non-linear dynamic model of a hot water heating system with feedback control has been added and results under open-loop conditions are presented. The model has the advantage of simplicity and computational efficiency. Results are compared with field-monitored data obtained from a building in use and an excellent agreement between the two is demonstrated. It is concluded that the model provides an excellent vehicle for short timescale investigations of HVAC plant and control.

2- INTRODUCTION

This work concerns itself with the modeling of building spaces and HVAC plant with specific reference to advanced controller design. Recent advances in HVAC control have increased the need for a thorough understanding of the dynamics of building spaces and associated plant. This is best achieved through the dynamic thermal modeling of the actual process involved.

Wiltshire and Wright comment that a thermal model can be "viewed as formal description of the behavior of a building and so require a description of all energy paths" [1]. A common theme is that all simulation models solve equations, which describe the conduction, convection, and radiant heat exchange in a building [2]. However, these mathematical models do not always replicate reality mainly because they are based on various assumptions and approximations. They are regarded as valid over some specified set of conditions [3].

Numerous computer simulation programs have been developed for building energy analysis such as DOE-2 [4] and TRNSYS [5]. Also, many simplified programs are simply computer implementations of handbook methods (e.g. ASHRAE [6]). Building energy analysis software has made a major contribution to advancing the precision, quality and productivity of building and HVAC system thermal design. However, with the exception of several open architecture programs such as TRNSYS [5], they are essentially black boxes providing the user with the restricted capability to specify inputs and sometimes output based on a set of predefined mathematical models of building and equipment components.

Usually, these models cannot be changed, thereby reducing the flexibility of design. Thus a major disadvantage of such software is that they are often used for conditions for which they are not valid, or their results are misinterpreted due to poor understanding of the mathematical models on which they are based. Also, most established thermal modeling programs are not suited to short-time scale investigations, the use of steady-state plant component models in some instances makes them prone to numerical instability when high-frequency input excitations are applied, and programs using finite-differences for building dynamics tend to be computationally inefficient.

In response to these shortcomings, what is needed is a simple modeling method which is numerically stable when dealing with a wide spectrum of plant input excitations and yet is computationally efficient. For the time horizons of interest in controller design, it is necessary and sufficient for such a modeling method to yield accurate results over several hours.

The aim of this paper is to develop and validate a flexible thermal model of a building and its heating system using Matlab and Simulink [7], where many control strategies may be applied and tested in order to investigate their behavior under realistic plant operating conditions.

3- MODEL DEVELOPMENT

3.1-BUILDING's MODEL

The factors which, influence the energy balance on a building space are: microclimate (external air temperature, wind effects and solar heat gains), casual heat gains, and heating systems as shown in figure (1).

Figure (2) illustrates the relationship between the various energy transfer paths in a building space with 2 external walls. From a control system perspective, we define the zone temperature (T_i) as a process output that has to be controlled. The controllable influence over zone air temperature (T_i) lies in one or more heating

system parameters–hot water flow rate or temperature (and, conceivably, both). Thus hot water flow rate and/or flow temperature are taken to be manipulated variables, and other inputs are interpreted as a measurable disturbances.

Any construction element consists of L layers of material which can be combined to form two "lumped" thermal resistances (R_{ins}, R_{out}), and one thermal capacity (C_{total}), as illustrated in figure (4).

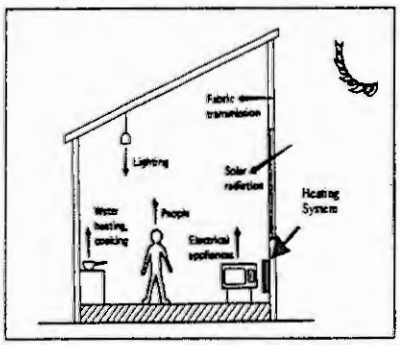


Figure (1)

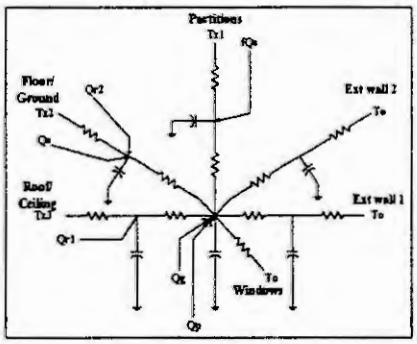


Figure (2)

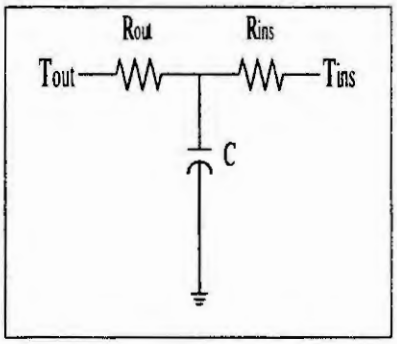


Figure (3)

The total thermal resistance and the total thermal capacitance can be calculated by following equations:

$$R_{total} = A_{total} \times (r_{si} + r_{so} + r_a + \sum_{1}^{L} \frac{x_l}{k_l}) \text{ and } C_{total} = A_{total} \times (\sum_{1}^{L} x_l \times \rho_l \times C_{pl}) \quad (1)$$

R_{ins} and R_{out} can be calculated by following equations, using the method prescribed by Lorenz and Masy [8]:

$$R_{ins} = \alpha \times R_{total} \text{ and } R_{out} = (1 - \alpha) \times R_{total} \quad (2)$$

α can be calculated for the external construction elements by following equations:

$$\alpha = 1 - [\frac{\sum_{1}^{L} R_k^{\bullet} \times C_k}{R_{total} \times C_{total}}] \quad (3)$$

$$\text{Where } R_k^{\bullet} = \sum_{1}^{L-1} R_i + \frac{R_k}{2} \quad (4)$$

Hence problem-specific state equations can now be written for a given building space, or group of spaces. This will be demonstrated later by way of a case example.

3.2- HEATING SYSTEM MODEL

The mathematical description of the heating system is divided into two parts. The first part is the description of the heat exchanger and its water connections, and the second part is a description of the control valve. Three possible control strategies exist, variable flow-rate constant temperature (VFCT), constant flow-rate variable temperature (CFVT), and variable flow-rate variable temperature (VFVT). Other than in special industrial applications, the type of hot water heat emitter most commonly used in current practice will provide natural convection or some combination of natural convection and radiation. Figures (4) shows the encased natural convector. An energy balance on the water-side gives:

$$C_h \dot{T}_{wo} = m \times C_{pw} \times (T_{wi} - T_{wo}) - Q_i \quad \text{and} \quad Q_w = h_{cw} \times A_w \times (T_{wo} - T_m) \quad (5)$$

And, an energy balance about the heat emitter material gives:

$$C_m \dot{T}_m = Q_w - Q_p \quad (6)$$

In which the heat emitted from the convector is assumed instantaneous and given by:

$$Q_p = h \times A_a \times (T_m - T_i)^n \quad (7)$$

For flow in tubes, the transition from laminar to turbulent flow occurs at [9]:

$$Re_d = \frac{\rho_w \times u_w \times d_i}{\mu_w} > 2300 \quad (8)$$

In which Re_d is the Reynolds number with respect to flow in tubes. It may also be written as:

$$\mathrm{Re_d} = \frac{m \times d_i}{A_{tcross} \times \mu_w} \tag{9}$$

Under turbulent flow conditions, the following empirical expression may be used to obtain the water-side convection heat transfer coefficient [9]:

$$\mathrm{Nu_d} = \frac{h_{cw} \times d_i}{k_w} = 0.023 \times \mathrm{Re_d^{0.8}} \times \mathrm{Pr^{0.33}} \tag{10}$$

In which $\mathrm{Nu_d}$ is Nusselts number for flow in tubes. Whereas for laminar flow ($\mathrm{Re_d} < 2000$) Nusselts number can be shown to be a constant value which for tubes is approximately 4.36, i.e.,

$$\mathrm{Nu_d} = \frac{h_{cw} \times d_i}{k_w} \cong 4.36 \tag{11}$$

In the transition region ($2000 < \mathrm{Re_d} < 4000$) Nusselts number (and, hence h_{cw}) may be interpolated with reference to the extremes defined by equations (10) and (11), i.e.,

$$4.36 \le \frac{h_{cw} \times d_i}{k_w} \le 0.023 \times \mathrm{Re_d^{0.8}} \times \mathrm{Pr^{0.33}} \tag{12}$$

3.2.1- CONTROL VALVE

Mathematical models of control valves for liquids are generally based on expressing the relationship between the flow rate passed by the valve and the position of the valve stem; the valve characteristic [10]:

$$G_{ins} = \left[1 + N(1/G_{inh}^2 - 1)\right]^{-1/2} \tag{13}$$

Where: $G_{inh} = G_0^{(1-u)}$ and $m_w = m \times G_{ins}$, ($G_0 = 0.005$ and $0.3 \le N \le 0.5$)

Thus the non-linear characteristic of the valve (e.g. figure (5)) attempts to compensate for the non-linear heat emission characteristic-equation (7). The simulation model was implemented using Matlab/Simulink. A block diagram representation of the heating system based on the open loop is shown in figure (6).

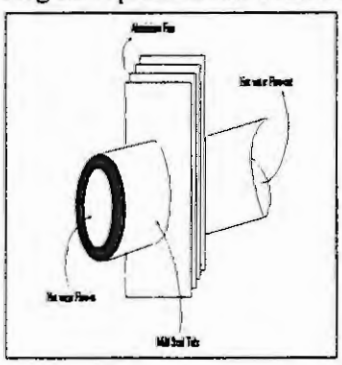


Figure (4)

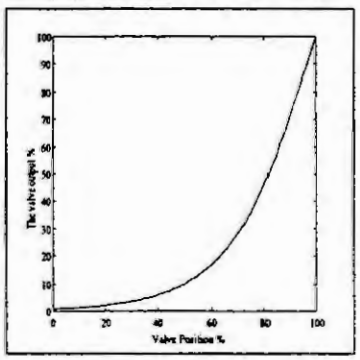


Figure (5)

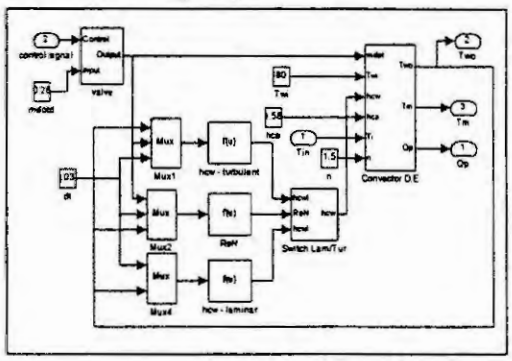


Figure (6)

4- MODEL APPLICATION

The model was applied to an example space in a campus building at the University of Northumbria. This building has been the subject of extensive research into photovoltaic cladding and is thus well documented in the literature [11]. The model of this building is shown in figure (7). The state space model for the thermal behavior of the building space, can be written as follows:

$$\dot{X} = AX + BU$$
$$Y = CX + DU \tag{14}$$

Where: X is the state vector: $[T_1\ T_2\ T_3\ T_4\ T_5\ T_i]^T$; U is the input vector: $[T_o\ Q_s\ Q_p\ Q_g\ Q_r\ T_{z1}\ T_{z2}\ T_{z3}]^T$ and Y the output vector (i.e. in this case the indoor space temperature, T_i). And $D = 0$.

5- MODEL IMPLEMENTATION AND VALIDATION

Validation is essential for the improvement in the quality of a model [12], since it increases confidence in the predicted result. The increasing use of thermal models requires that their accuracy be assessed regularly. Validation is therefore an integral part of model development. The validation method has been applied here

based on a data sample for a period of several consecutive days from the (north-facing) example space when the heating was active but the room known to be unoccupied. The measured data sample consisted of a one-week (approximate) time series at 15-minute intervals of space dry bulb temperatures and heating control valve signals. The external temperature time-series was applied directly, and then filtered in order to smooth out some unacceptable noise spikes. The final model was implemented using the Matlab-Simulink realization shown in figure (8). In figure (9), the response to several cycles of heating is compared with field measurement data. The initial period represents a weekend whilst the 5 cycles of heating and cooling represent Monday-Friday. Results based on the model and field-measurements compare very favourably after the external temperature time series had been filtered to remove noise from the data.

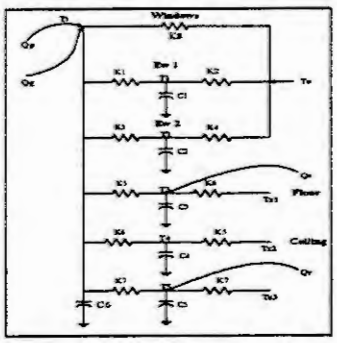


Figure (7)

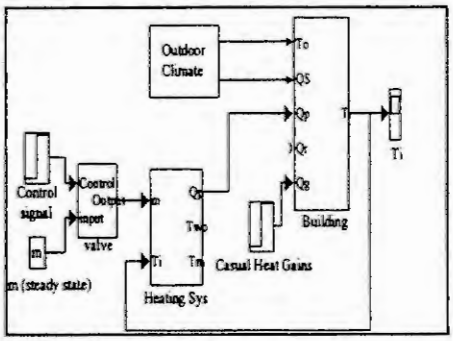


Figure (8)

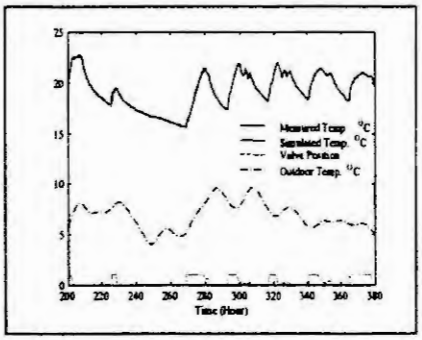


Figure (9)

6- CONCLUSION

In this work, the development of a simplified approach to modeling a building space and heating system has been investigated for the specific purpose of the short time-scale simulation of relevance to control system synthesis and design. A building is a complex process to control, due to the large time delays, time variant parameters, and plant uncertainties involved.

The model developed in this work represents the building as a low-order linear state-space realization with a lumped-parameter non-linear heating system description. The resulting model has been implemented using Simulink in the Matlab environment. Excellent agreement has been found between the model and results from field monitoring of a building exhibiting high thermal capacity characteristics. It is concluded that the model is simple, computationally efficient and sufficiently accurate to be particularly suited to short time-scale simulations appropriate to control system analysis.

7- REFERENCES

1- Wiltshire, T. J., Wright, A. J. "Advances in building energy simulation in the UK: the Science and Engineering Research Council's Program" *Energy and Buildings,* 10, pp175-183, 1988.

2- Lomas, K. J., Bowman, N. T. "The evaluation and use of existing data sets for validation dynamic thermal models of buildings" *Proc. 5th Int. Symp. Use of Computers for Environmental Engineering Related to Buildings*, pp99-107, 1986.

3- Irving, A. D. "Validation of dynamic thermal models" *Energy and Buildings*, 10, pp213-220, 1988.

4- *DOE-2 Supplement Ver 2. ID. Lawrence Berkley Laboratory Report N. LBL-8706 Rev. 5.* DOE-2 User Coordination Office, LBL, Berkley, Ca., 1989.

5- *TRNSYS: A Transient System Simulation Program*, Solar Energy Laboratory, University of Wisconsin, Madison, WI, 1990.

6- *ASHRAE Handbook-Fundamentals*, Atlanta, Georgia, U.S.A., 1993.

7- *Matlab*, MathWorks Inc., Natvick, Mass., U.S.A., 1993.

8- Lorenz, F., Masy, G. "Methode d'evaluation de l'economie d'energie apportee par l'intermittence de chauffage dans les batiments. Traitement par differences finies d'un model a deux constantes de temps" *Report No. GM820130-01*, University de Liege, Faculte des Sciences Appliquees, 1982 (in French).

9- Holman, J. P. "*Heat Transfer*", McGraw Hill, New York, 1997.

10- Underwood, C. P., Edge, J. S. "Flow characteristics in circuits using three-port modulating *control" Building Services Engineering Research and Technology* 16 (3), pp127-132, 1995.

11- Horne, M. "Modeling photovoltaic-clad facades". *Building Services Engineering Research & Technology,* 19 (4), ppB10-B12, 1998.

12- Clark, J. "Computer Application in the design of energy conscious" buildings Computer *Aided Design* 14 (1), pp3-9, 1982.

MODELLING HEAT TRANSFER BETWEEN SOLID PARTICLES

Cs. Mihálykó[1], B. G. Lakatos[1] and T. Blickle[2]
[1]University of Veszprém, H-8201 Veszprém, PO Box 158, Hungary
[2]Research Institute of Chemical and Process Engineering, Pannon University of Agriculture
H-8201 Veszprém, PO Box 125, Hungary

Abstract. A population balance model is presented for describing the direct particle-to-particle heat transfer in fluid-solid processing systems. In developing the model, simple kinetic equations are assumed to describe the heat transfer process, but the particle-particle interactions are considered stochastic. A new approach is used to develop the population balance equation. The moment equations are derived and analysed. The time evolution of the population density function, determined by numerical solution of the integro-differential equation, is also presented.

Introduction

In modelling fluid-solid processing systems, usually two different methods are used for describing heat transfer processes. For dilute systems, the discrete particle simulation approach handles each particle separately [1-2], while the pseudo-homogeneous formulation treats the particulate phase as a continuous one [3-4]. Mostly neither of these methods considers the direct particle-to-particle heat transfer. By using the first method, this transfer process is often neglected because of the modelling difficulties [2], while in the pseudo-homogeneous formulation all particle interactions are aggregated into some continuous parameters. However, by using the population balance approach this process may be modelled and separated from the fluid-particle and bed-to-suspension heat transfer processes.

The aim of the paper is to present a population balance model for describing the particle-to-particle heat transfer. The particle interactions are considered stochastic, and a new approach is used to develop the model. The moment equations and numerical solution are also presented and analysed.

Heat transfer kinetics

Consider a large population of particles having identical sizes and physical properties, moving and collide in some fluid medium. Let us assume that this material system is closed and isolated from the environment; all properties of particles are constant during their motion expect the temperature; the number of particles in the system is constant; the heat transfer through the fluid medium is negligible; the temperature inside each particle is homogeneous; then the only process changing the temperature of particles is their direct contact heat transfer caused by collisions.

Consider two particles having, according to the former conditions, identical mass m and heat capacity C, but different temperatures T_{10} and T_{20}, respectively. If these particles are collide and remain in contact for some time θ, then equalising heat transfer occurs between them, and their temperatures become T_1 and T_2, respectively. Assuming that this heat transfer can be described by an overall heat transfer coefficient β, then the equations describing the variation of temperatures of particles are

$$mC\frac{dT_1}{dt} = \beta A(T_2 - T_1) \text{ and } mC\frac{dT_2}{dt} = -\beta A(T_2 - T_1) \tag{1}$$

subject to the initial conditions $T_1(0) = T_{10}, T_2(0) = T_{20}$, where A is the heat transfer (contact) area.

If the particles remain in contact to time θ, then the solution of the set of linear differential equations at time θ gives the final temperatures of particles

$$T_1(\theta) = T_{10} + \frac{(T_{20} - T_{10})}{2}\left[1 - \exp\left(\frac{-\beta A\theta}{mC}\right)\right] \text{ and } T_2(\theta) = T_{20} - \frac{(T_{20} - T_{10})}{2}\left[1 - \exp\left(\frac{-\beta A\theta}{mC}\right)\right]. \tag{2}$$

In Eqs (1) and (2), parameters β, A, θ are, in essence, random quantities since the quality and area of contact, as well as the contact time in such system may depend on a number of random conditions. As a consequence, parameter $\omega = 1 - \exp\left(\frac{-\beta A\theta}{mC}\right)$ is a random function of the quantities β, A, θ, and its distribution is entirely determined by the distribution functions of parameters β, A, θ. Naturally, $\omega \in [0,1]$. Furthermore, we

suppose that the parameters β, A, δ are independent, hence the distribution of ω can be determined by means of the well-known formulae. We suppose that the density function b of the distribution of ω is known.

The population balance equation

Now we show a new method to set up the population balance equation. Let us assume that the total number of particles in the system is N, and $N(.,.)$ denotes the population distribution function of the particles. Here, $N(T,t)$ gives the number of particles at time t the temperature of which is less then T. As a consequence, $F_N(T,t) = \frac{N(T,t)}{N}$ is the normalised number distribution function of the particle population. Let N be such a large number, that $F_N(T,t)$ can be approximated (in T and t uniformly) satisfactorily by a family of distribution functions $F(T,t)$ the members of which are differentiable with respect to T and t. Let $f(T,t)$ be $\frac{\partial F}{\partial T}(T,t)$. If now ξ_t denotes the temperature of a randomly chosen particle of the population at time t, then the distribution function of ξ_t is $F_N(.,t)$, thus $F(.,t)$ can be substituted for it. Furthermore, we consider the difference between $F_N(.,t)$ and $F(.,t)$ negligible, thus $F(.,t)$ and $f(.,t)$ are taken as the distribution and density functions of ξ_t, respectively.

Since the heat transfer between the particles occurs only by particle contacts, that is the effects of the fluid medium are neglected, the changes in the temperature of a particle depend only on two conditions: if the given particle is contacted with another particle of different temperature, and what are the actual values of the parameters characterising the heat transfer process. Now let us suppose that the probability that the temperature T_1 of the particle changes in the interval of time $(t,t+\tau)$ (if $\tau \ll 1$ is sufficiently small) under the condition that it meets only one particle of temperature $T_2 \neq T_1$ and the heat transfer process proceeds with parameter ω is $k\tau + o(\tau)$, independently of t, T_1, T_2 and ω, where $k \in [0,1]$. Further, we suppose that the probability that one particle takes part in more than one heat transfer process during this time is $o(\tau)$.

Let we choose randomly one particle at time $t+\tau$ from the population. If it is such a particle that met at most one other particle in the interval of time $(t,t+\tau)$ and its temperature did not change during that time then this event is denoted A_1. If it is such a particle that met exactly one other particle in the interval of time $(t,t+\tau)$ and its temperature changed during that time then this event is denoted A_2. Finally, let A_3 be the complement of $\bigcup_{i=1}^{2} A_i$. Now the temperature $\xi_{t+\tau}$ of the observed particle becomes (using Eqs (1) and (2)): if A_1 occurs, then $\xi_{t+\tau} = \xi_{1,t}$; if A_2 occurs, then $\xi_{t+\tau} = \xi_{1,t} + (\xi_{2,t} - \xi_{1,t})\frac{\xi_3}{2}$; finally, we do not need the explicit form of $\xi_{t+\tau}$ when A_3 occurs since $P(A_3) = o(\tau)$. Therefore, we denote it simply by $\xi_{t+\tau}^{(3)}$. Now $\xi_{1,t}$ denotes the temperature of the particle at time t which was chosen, $\xi_{2,t}$ denotes the temperature of the particle at time t, that the chosen particle met, and ξ_3 denotes the value of ω characterising the heat transfer process between the particles. We suppose that N is large enough to consider $\xi_{1,t}$ and $\xi_{2,t}$ independent, identically distributed random variables and their distribution is the same as the distribution of ξ_t. Finally we suppose that ξ_3 is also independent of $\xi_{1,t}$ and $\xi_{2,t}$ the density function of which is $b(.)$.

Then $\xi_{t+\tau} = \xi_{1,t} \cdot 1_{A_1} + (\xi_{1,t} + (\xi_{2,t} - \xi_{1,t})\frac{\xi_3}{2}) \cdot 1_{A_2} + \xi_{t+\tau}^{(3)} \cdot 1_{A_3}$ where 1_{A_k}, k=1,2,3 are the characteristic functions of the sets A_k, k=1,2,3. According to these considerations, we can determine the joint probability density function of variables $(\xi_{2,t}, \xi_3, \xi_{t+\tau})$. Then, expressing the marginal probability density function $f(T,t+\tau)$ of the variable $\xi_{t+\tau}$, which has the property that $f(T,.) = 0$, if $T \notin [T_{min}, T_{max}]$, we obtain the following equation for $f(T,t+\tau)$

$$f(T,t+\tau)=\int_{T_{min}}^{T_{max}}\int_{0}^{1} f(T,t)f(S,t)b(\omega)[1-k\tau-2o(\tau)]d\omega dS+$$
$$+\int_{T_{min}}^{T_{max}}\int_{0}^{1} f(S,t)f\left(\frac{2(T-S)}{\omega}+S,t\right)\frac{2}{\omega}b(\omega)[k\tau+o(\tau)]d\omega dS+o(\tau) \tag{3}$$

Introducing the notation $n(T,t)=Nf(T,t)$ where $n(T,t)dT$ expresses the number of particles with temperature from the interval $(T,T+dT)$, we get the following integro-differential equation as $\tau\to 0$:

$$\frac{\partial n}{\partial t}(T,t)=\frac{k}{N}\left(-n(T,t)\int_{T_{min}}^{T_{max}} n(S,t)dS+\int_{T_{min}}^{T_{max}}\int_{0}^{1} n(S,t)n\left(\frac{2(T-S)}{\omega}+S,t\right)\frac{2}{\omega}b(\omega)d\omega dS,\quad t>0\right. \tag{4}$$
$$n(T,0)=n_0(T),\quad T\in[T_{min},T_{max}]$$

This equation is, in essence, the population balance equation of the heat transfer process under the above mentioned simplifying conditions and it describes the evolution of the population density function of the particles in time.

The moment equations

The moments of the population density function $n(.,.)$, expressed as

$$M_I(t)=\int_{T_{min}}^{T_{max}} T^I n(T,t)dT,\quad I=0,1,2,... \tag{5}$$

often provide very useful information about the system, especially when the integro-differential equation (4) may not be solved explicitly.

Multiplying both sides of Eq.(4) by T^I and integrating from T_{min} to T_{max}, after some suitable transformations we get the following system of ordinary differential equations:

$$\frac{dM_I(t)}{dt}=\frac{k}{N}\left(-M_I(t)M_0(t)+\sum_{i=0}^{I} M_i(t)M_{I-i}(t)b_{i,I}\right),\quad I=0,1,2,...\quad t>0 \tag{6}$$
$$M_I(0)=M_{I,0}$$

where

$$b_{i,I}=\int_0^1\binom{I}{i}\left(\frac{\omega}{2}\right)^i\left(1-\frac{\omega}{2}\right)^{I-i}b(\omega)d\omega. \tag{7}$$

We note that $\sum_{i=0}^{I} b_{i,I}=1$ and $b_{i,I}\geq 0$. Furthermore, in the case of $i\geq 1$ $b_{i,I}=0$ if and only if $b(\omega)$ is Dirac-delta function at $\omega=0$. Eqs (6) form a recursive set of ordinary equations, i.e. if we know $M_0(t), M_1(t),..., M_{I-1}(t)$, then Eq.(6) is a linear inhomogeneous differential equation for $M_I(t)$.

When $I=0$ we get $dM_0(t)/dt=0$ that is $M_0(t)\equiv M_{0,0}(=N)$, what is expected, since it means that the total number of particles is constant. When $I=1$ we get $dM_1(t)/dt=0$ that is $M_1(t)\equiv M_{1,0}$. This means that during the heat transfer process the total amount of heat of the system also remains constant according to the expectations. This later consequence together with the case $I=0$ shows that our model is an adequate model of the physical process. In the case of $I=2$

$$\frac{dM_2(t)}{dt}=-kb_{1,2}\left(M_2(t)-\frac{M_{1,0}^2}{M_{0,0}}\right) \tag{8}$$

thus we get

$$M_2(t)=\left(M_{2,0}-\frac{M_{1,0}^2}{M_{0,0}}\right)e^{-kb_{1,2}t}+\frac{M_{1,0}^2}{M_{0,0}}. \tag{9}$$

Using this expression, we can calculate the standard deviation, namely

$$\sigma^2(t)=\frac{M_2(t)}{M_0(t)}-\frac{M_1^2(t)}{M_0^2(t)}=\left(\frac{M_{2,0}}{M_{0,0}}-\frac{M_{1,0}^2}{M_{0,0}^2}\right)e^{-kb_{1,2}t}=\sigma^2(0)e^{-kb_{1,2}t}. \tag{10}$$

Taking into account our former remark with respect to $b_{i,l}$ we see that if $t \to \infty$ then $\sigma^2(t) \to 0$ except the case when $b(\omega)$ is a Dirac-delta function at $\omega = 0$. This is the special case when there is no heat transfer between the particles at all. In general, the model predicts that the temperature distribution of the particle population equalises with time and becomes uniform as $t \to \infty$. As a consequence, the moments got from the population balance equation predicts an adequate behaviour of the system.

Numerical results

A numerical method, based on approximating the time derivative by a forward difference scheme and the integral by a quadrature formula, has been elaborated for solving the integro-differential equation (4). The details of the method and its properties will be published elsewhere. Fig.1 presents the evolution of the population density function for the reduced interval $[T_{min}, T_{max}] = [0,1]$. The initial density function was $n(T,0) \equiv 1$, while the density function of parameter ω was taken uniform $b(\omega) \equiv 1$. It is observed that the distribution of the temperature is concentrated on the average value of the temperature $(M_{1,0} = 0.5)$ as the time passes.

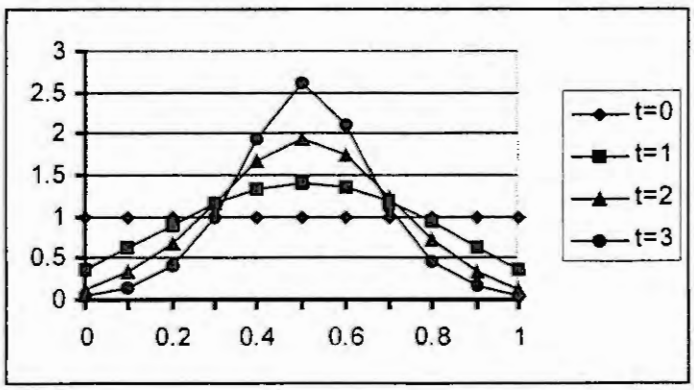


Figure 1. Evolution of the population density function $n(T,t)$ in time

Conclusions

The population balance model presented and analysed in the paper seems to be rather promising for describing the direct particle-to-particle heat transfer in fluid-solid processing systems. By using this model, we are able to take into account a number of parameters affecting the process, and it provides a means to describe the local temperature inhomogeneities of the particles. This is especially important in the case of the highly exothermic processes in which hot pots may appear because of the not satisfactory mixing of the particulate phase. In further development of the model, the fluid-particle heat transfer will also be considered, and by means of such generalisation we will obtain population balance-based models for a number of important fluid-solid systems.

Acknowlegdement. The authors thank dr. László Koltay for his valuable comments. The work was supported by the Hungarian Research Foundation under Grants *F023529* and *T26228*, and by the Ministry of Culture and Education of Hungary under Grant *FKFP 0520/1999*. The financial support is gratefully acknowledged.

References

1. Tsuji, Y., Kawaguchi, T. and Tanaka, T., Discrete particle simulation of two-dimensional fluidized bed. Powder Technology, 77 (1993) 79-87.
2. Kaneko, Y., Shiojima, T. and Horio, M., DEM simulation of fluidized beds for gas-phase olefin polymerization. Chem.Engng Sci., 54 (1999) 5809-5881.
3. Kuipers, J.A., Prims, M. and van Swaaij, W.P.M., Numerical calculation of wall-to-bed heat transfer coefficients in gas-fluidized beds. AIChE Journal, 38 (1992) 1079-1091.
4. Schmidt, A. and Renz, U., Eulerian computation of heat transfer in fluidized beds. Chem.Engng Sci., 54 (1999) 5515-5522.

Modeling and Simulation of a Hydrostatic Transmission with Variable-Displacement Pump

Andreas Kugi[1], **Kurt Schlacher**[1], **Heinz Aitzetmüller**[2] and **Gottfried Hirmann**[2]
[1]Department of Automatic Control, Johannes Kepler University of Linz
Altenbergerstraße 69, A-4040 Linz, AUSTRIA
e-mail: andi@regpro.mechatronik.uni-linz.ac.at
[2]Steyr-Daimler-Puch AG, Antriebstechnik
Schönauerstr. 5, A-4400 Steyr, AUSTRIA

Abstract. The S-matic Power Split Drive of Steyr Antriebstechnik is a drive box for vehicular drive systems which combines the advantages of hydrostatic and mechanical transmission. This paper is concerned with the mathematical modeling of the hydrostatic unit of the drive box with special emphasis on the swash-plate mechanism of the variable-displacement pump. For reliability reasons no measurement device is planned for the variable swash-plate angle. Therefore, a simple discrete on-line simulator for the swash-plate angle is derived by gradually simplifying the mathematical model on the basis of physical considerations.

Introduction

In order to meet the increasing demands on vehicular drive systems regarding economy, environmental influence and comfort, Steyr Antriebstechnik developed the S-matic Power Split Drive. This drive box consists of a hydrostatic unit with a variable-displacement pump and planetary gears in combination with dog clutches. The incoming engine power is split inside the drive box into a mechanical and a hydrostatic part. This concept combines the advantages of both, hydrostatic transmission, which enables a continuously variable speed under all working conditions, and mechanical transmission, which is responsible for a high efficiency and durability in combination with the avoidance of stick-slip-effects in low-speed conditions.

In this paper we focus our attention on the derivation of the mathematical model of the hydrostatic unit of the S-matic Power Split Drive with regard to control applications. As it is mentioned in [3], most of the traditionally used models for hydrostatic transmission neglect essential parts of the dynamic behavior of the system, specifically the dynamics of the swash-plate and the corresponding control unit of the variable-displacement pump. The main reason for this is that mostly these models are used for a static or quasi-static analysis of the hydrostatic transmission. In the following, we present a mathematical model comprising the equations of motion for the hydrostatic motor and pump, respectively, as well as the oil flow equations for the high- and low-pressure sides. Special emphasis is laid on a detailed mathematical formulation of the swash-plate mechanism for the variable-displacement pump. The swash-plate angle is a very important information for the drive control unit because it determines the displacement of the pump and hence the speed of the hydrostatic motor. For reliability reasons no measurement device is planned for the swash-plate angle within the S-matic Power Split Drive and, therefore, an on-line simulator has to be implemented. Since the mathematical model of the swash-plate mechanism is highly complex and the simulator has to be implemented in the hardware platform of the drive unit system, which only allows a limited sampling time, the complexity of the model is gradually reduced on the basis of physical considerations. The obtained on-line simulator also shows a good insensitivity to variations in the parameters of the constitutive equations. A comparison between simulation and experimental results will demonstrate that the proposed simulator optimally fits the reality and is practically feasible.

Mathematical model of the pump-motor-unit

Since the transmission lines, which connect the pump and the motor, are short we can represent the low- and high-pressure side (transmission line + chambers in the pump or motor + connecting passages) by one pressure value p_1 and p_2, either. The hydrostatic unit is equipped with a flushing and a boost system where the boost pump is supposed to ideally compensate for the drained oil from the flush valve.

Henceforth, we will assume that the density of oil ρ_{oil} is independent of the temperature T. But it is a well known fact that the dynamic viscosity of oil $\mu(T)$ changes markedly with the temperature T. This is why, we will take the dependence of the dynamic viscosity on the temperature into consideration but we assume that the change of the temperature $\mathrm{d}T/\mathrm{d}t$ is so slow that it is negligible. Then, by using for the bulk modulus of oil β_{oil} the definition $\beta_{oil} = \rho_{oil}\,(\partial p/\partial \rho_{oil})_{T=\text{const.}}$ [5], the mathematical model for the pump-motor-unit can be described in the form

$$\begin{aligned}
\frac{V_1}{\beta_{oil}}\frac{\mathrm{d}}{\mathrm{d}t}p_1 &= \frac{N_p}{2\pi}A_pD_p\tan(\alpha_p)\,\omega_p - \frac{N_m}{2\pi}A_mD_m\tan(\alpha_m)\,\omega_m - \frac{C_{int}}{\mu(T)}(p_1-p_2) - \frac{C_{ext,1}}{\mu(T)}p_1\\
\frac{V_2}{\beta_{oil}}\frac{\mathrm{d}}{\mathrm{d}t}p_2 &= \frac{N_m}{2\pi}A_mD_m\tan(\alpha_m)\,\omega_m - \frac{N_p}{2\pi}A_pD_p\tan(\alpha_p)\,\omega_p + \frac{C_{int}}{\mu(T)}(p_1-p_2) - \frac{C_{ext,2}}{\mu(T)}p_2\\
\Theta_p\frac{\mathrm{d}}{\mathrm{d}t}\omega_p &= M_{drive} - \frac{N_p}{2\pi}A_pD_p\tan(\alpha_p)\,(p_1-p_2) - k_{d,p}\mu(T)\,\omega_p - k_{p,p}\,(p_1+p_2)\,\mathrm{sign}(\omega_p)\\
\Theta_m\frac{\mathrm{d}}{\mathrm{d}t}\omega_m &= \frac{N_m}{2\pi}A_mD_m\tan(\alpha_m)\,(p_1-p_2) - M_{load} - k_{d,m}\mu(T)\,\omega_m - k_{p,m}\,(p_1+p_2)\,\mathrm{sign}(\omega_m)
\end{aligned}$$

with the total volumes (transmission line + chambers + connecting passages) of the two pressure sides V_1 and V_2, the angular velocity ω, the maximum geometric displacement D of the pistons, the effective piston area A, the number of pistons N, the moment of inertia Θ, the stroke angle α, the drive torque of the pump M_{drive} and the load torque of the motor M_{load}. Further, $C_{ext,1}$, $C_{ext,2}$ and C_{int} denote the external and internal leakage coefficients and k_d and k_p are friction parameters [1], [5]. Here and subsequently, an index p or m always refers to the corresponding quantity of the pump or motor, respectively.

Mathematical model of the swash-plate mechanism

An excellent description of the closed-form equations of a variable-displacement pump can be found in [3] and [4]. In contrast to our approach, the main attention there is directed to the design problem and the dynamics of the swash-plate mechanism under parameter variations. Fig. 1, left hand side, depicts the schematic diagram of the swash-plate mechanism of the variable-displacement pump. By means of the two piston forces $F_A = p_A A_A$, $F_B = p_B A_B$ and the spring force F_S the swash-plate angle α_p can be controlled in a range $-\alpha_{p,\max} \le \alpha_p \le \alpha_{p,\max}$. Calculating the rate of change of the angular momentum around the swash-plate pivot, we get the equations of motion in the form

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\alpha_p &= \nu_p\\
\frac{\mathrm{d}}{\mathrm{d}t}\nu_p &= \frac{R_{S,eff}\cos(\alpha_p)\,\left(F_A - F_B - F_S + (m_{S,A}+m_{S,B})\,R_{S,eff}\sin(\alpha_p)\,\nu_p^2\right) - M_{S,fric} - M_S}{\left(\Theta_S + (m_{S,A}+m_{S,B})\,R_{S,eff}^2\cos^2(\alpha_p)\right)}
\end{aligned} \tag{1}$$

with $F_S = F_{pre} + c_S R_{S,eff}\,(\sin(\alpha_p) + \sin(\alpha_{p,\max}))$ for $-\alpha_{p,\max} \le \alpha_p \le \alpha_{p,\max}$. Here, Θ_S denotes the moment of inertia of the swash-plate, $M_{S,fric}$ is the friction torque, M_S is the so called swivel torque, which is naturally induced by the pump [3], [4], $m_{S,A}$ and $m_{S,B}$ denote the sum of the masses of the piston and the piston rod of the two actuators A and B, respectively, and F_{pre} stands for the prestress-force of the restoring spring and c_S is the spring coefficient. The actuators A and B of fig. 1, left hand side, are single-ended, single-acting hydraulic rams. The continuity equations for the two chambers read as

$$\begin{aligned}
\frac{(V_{0,A} + A_A R_{S,eff}\sin(\alpha_p))}{\beta_{oil}}\frac{\mathrm{d}}{\mathrm{d}t}p_A &= q_A - A_A R_{S,eff}\cos(\alpha_p)\,\nu_p - \frac{C_{ext,A}}{\mu(T)}p_A\\
\frac{(V_{0,B} - A_B R_{S,eff}\sin(\alpha_p))}{\beta_{oil}}\frac{\mathrm{d}}{\mathrm{d}t}p_B &= q_B + A_B R_{S,eff}\cos(\alpha_p)\,\nu_p - \frac{C_{ext,B}}{\mu(T)}p_B
\end{aligned} \tag{2}$$

with the pressures p_A and p_B in the two chambers, the effective piston areas A_A and A_B, the volumes $V_{0,A}$ and $V_{0,B}$ for $\alpha_p = 0$ and the external leakage coefficients $C_{ext,A}$ and $C_{ext,B}$. The flows from and to the valves of the two chambers, q_A and q_B, are determined by a hydro-mechanical feedback mechanism in such a way that the error $\Delta\alpha_p = \alpha_p - \alpha_{p,d}$ between the actual and the desired swash-plate angle, α_p and $\alpha_{p,d}$, becomes zero. This is achieved by connecting the corresponding chamber via an orifice area

$A_o(\Delta\alpha_p)$ with the tank or supply pressure p_T and p_S, respectively. Thus, the flows are given by

$$\Delta\alpha_p > 0: \begin{cases} q_A = C_d\sqrt{\frac{2}{\rho_{oil}}}A_o(\Delta\alpha_p)\sqrt{p_S - p_A} \\ q_B = -C_d\sqrt{\frac{2}{\rho_{oil}}}A_o(\Delta\alpha_p)\sqrt{p_B - p_T} \end{cases}, \Delta\alpha_p < 0: \begin{cases} q_A = -C_d\sqrt{\frac{2}{\rho_{oil}}}A_o(\Delta\alpha_p)\sqrt{p_A - p_T} \\ q_B = C_d\sqrt{\frac{2}{\rho_{oil}}}A_o(\Delta\alpha_p)\sqrt{p_S - p_B}, \end{cases} \tag{3}$$

where C_d is the discharge coefficient. For slit-type sharp-edged orifices the discharge coefficient may be set $C_d \approx 0.6$, regardless of the particular geometry (see e.g., [5]).

On-line simulator for the swash-plate angle

In order to obtain a simple simulator, which captures the essential dynamics of the swash-plate mechanism, the complexity of the detailed mathematical model (1) - (3) will be gradually reduced on the basis of physical considerations. In a first step, let us assume that the two actuators A and B have the identical geometry, i.e., $A_A = A_B = A$ and $V_{0,A} = V_{0,B} = V_0$. Further, if we neglect the external leakage flows $C_{ext,A} = C_{ext,B} = 0$, then in the steady state the flows q_A and q_B from (3) follow the relation $q_A = -q_B$. Thus, with the assumption $p_T = 0$, we get $p_A + p_B = p_S$ and we are able to formulate (3) in terms of the load pressure $\Delta p = p_A - p_B$. Additional investigations have shown that a Taylor series approximation around $\Delta p = 0$ up to the first order of (3) suffices for the description of the dynamics of the system and thus we have

$$q_A = -q_B = C_d\sqrt{\frac{2}{\rho_{oil}}}A_o(\Delta\alpha_p)\left(\sqrt{p_S}\,\mathrm{sign}(\Delta\alpha_p) - \frac{1}{2\sqrt{p_S}}\Delta p\right). \tag{4}$$

Since in (2) the expressions on the left hand side, which is dominated by $1/\beta_{oil}$, are very small, we can replace the dynamic equations (2) by their quasi-steady-state representation in the sense of the singular pertubation theory (see e.g. [2]). Thus, the differential equations (2) degenerate into equations and by inserting (4) into these equations, we obtain an expression for the load pressure Δp in the form

$$\Delta p = -\frac{\sqrt{2\rho_{oil}p_S}AR_{S,eff}}{C_dA_o(\Delta\alpha_p)}\cos(\alpha_p)\nu_p + 2p_S\,\mathrm{sign}(\Delta\alpha_p). \tag{5}$$

In a second step, our analysis shows that in (1) the masses $m_{S,A}$ and $m_{S,B}$ of the piston and the piston rod of the two actuators A and B as well as the friction and the swivel torque $M_{S,fric}$ and M_S only make a minor contribution to the torque balance. Furthermore, in the considered operating range of the swash-plate angle $-21.5\,^\circ \le \alpha_p \le 21.5\,^\circ$ all expressions in α_p may be linearized around $\alpha_p = 0$, i.e. $\cos(\alpha_p) \cong 1$, $\sin(\alpha_p) \cong \alpha_p$ and $\sin(\alpha_{p,\max}) \cong \alpha_{p,\max}$. With these simplifications (1) becomes

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\alpha_p &= \nu_p \\ \Theta_S\frac{\mathrm{d}}{\mathrm{d}t}\nu_p &= R_{S,eff}\left(A\Delta p - F_{pre} - c_SR_{S,eff}(\alpha_p + \alpha_{p,\max})\right). \end{aligned} \tag{6}$$

In the same manner as for the hydraulic part, we will regard the moment of inertia of the swash-plate Θ_S in (6) as a pertubation parameter. Also in [4] and in the literature cited therein it is considered that the inertia of the swash-plate is negligible to the stiffness of the servosystem. Substituting (5) in the quasi-steady-state representation of (6) and taking into account that the expression $2Ap_S$ is much bigger than $-F_{pre} - c_SR_{S,eff}\alpha_{p,\max}$, we end up with

$$\frac{\mathrm{d}}{\mathrm{d}t}\alpha_p = \underbrace{-\frac{C_dA_o(\Delta\alpha_p)c_S}{A^2\sqrt{2\rho_{oil}p_S}}}_{\zeta_1}\alpha_p + \underbrace{\frac{C_dA_o(\Delta\alpha_p)}{AR_{S,eff}}\sqrt{\frac{2p_S}{\rho_{oil}}}}_{\zeta_2}\mathrm{sign}(\Delta\alpha_p). \tag{7}$$

Now, the continuous model of (7) serves as a basis for the discrete open-loop on-line simulator. Suppose that the quantities supply pressure p_S and orifice area $A_o(\Delta\alpha_p)$ can be considered to remain constant,

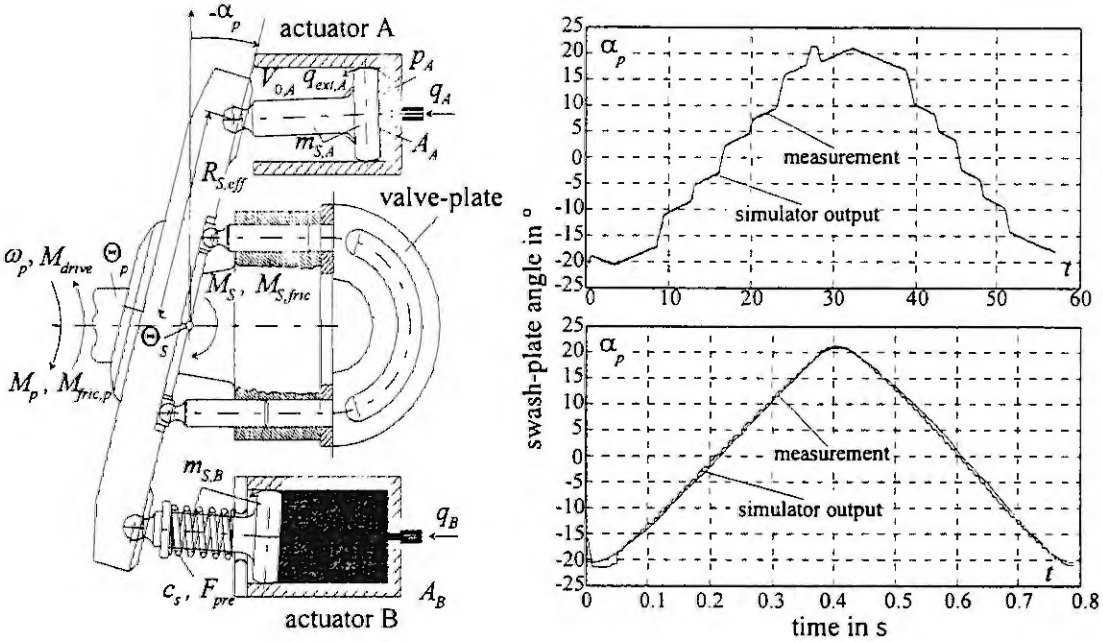


Figure 1: Left hand side: Schematic diagram of the swash-plate mechanism. Right hand side: Measured and observed swash-plate angle for a calibration process and for a quick swash-plate turn.

namely $p_{S,k}$ and $A_o(\Delta\alpha_{p,k})$, during the sampling intervals $kT_a \leq t < (k+1)T_a$, $k = 0, 1, 2, \ldots$ with the sampling time T_a. Then the corresponding exact discrete model to (7) can be easily calculated by

$$\alpha_{p,k+1} = \exp(\zeta_{1,k} T_a)\, \alpha_{p,k} + \frac{\zeta_{2,k}\left(\exp(\zeta_{1,k} T_a) - 1\right)}{\zeta_{1,k}} \operatorname{sign}(\Delta\alpha_{p,k}) \ . \tag{8}$$

Fig. 1, right hand side, presents the comparison results of the measured and the estimated swash-plate angle due to (8) for a calibration test and for a quick swash-plate turn with an initial error of 5 ° in the swash-plate angle of the S-matic Power Split Drive for a supply pressure $p_S = 28 \times 10^5$ Nm^{-2}, an average temperature $T = 60$ ° and a sampling time $T_a = 10 \times 10^{-3}$ s. The orifice area as a function of the swash-plate angle error $A_o(\Delta\alpha_p)$ was made available by the manufacturer of the swash-plate mechanism.

Conclusion

The paper presents a new discrete on-line simulator for the swash-plate angle of the variable-displacement pump of the hydrostatic unit within the S-matic Power Split Drive developed by Steyr Antriebstechnik.

References

1. Blackburn, J. F., Reethof, G., and Shearer, J. L., Fluid Power Control, John Wiley, 1960.
2. Khalil, H., Nonlinear Systems, Macmillan Publishing Company, 1992.
3. Manring, N. D. and Johnson, R. E., Modeling and Designing a Variable-Displacement Open-Loop Pump, Journal of Dynamic Systems, Measurement, and Control, 118 (1996), 267 – 271.
4. Manring, N. D. and Luecke, G. R., Modelling and Designing a Hydrostatic Transmission with a Fixed-Displacement Motor, Journal of Dynamic Systems, Measurement, and Control, 120 (1998), 45 – 49.
5. Merritt, H. E., Hydraulic Control Systems, John Wiley, 1967.

MODELLING OF PRESSURE FROM DISCHARGES AT ACTIVE WELLS BY SOIL VENTING FACILITIES

M. Slodička and H. De Schepper
Department of Mathematical Analysis, University of Ghent
Galglaan 2, B-9000 Gent, Belgium

Abstract. Soil venting is a commonly used technique for the remediation of the unsaturated zone of the soil, which is contaminated by gaseous organic pollutants present in the pores. We study a steady state model with a finite number of extraction wells. The air flow field in the subsurface can be described by a linear elliptic partial differential equation, accompanied of a *nonlocal* Neumann boundary condition, along with a Dirichlet side condition, containing *unknown* parameters. We prove the well–posedness of the problem and we develop a method for the parameter identification, i.e. for the determination of the pressure values at the active wells from their discharges.

Introduction

The infiltration of organic solvents into the subsurface leads to contamination of the soil and of the ground water system. In cases where volatile hydrocarbons enter the unsaturated zone, very often *soil venting* is used for remediation. This technique is used in practice by specialised firms, e.g, by GEO-data in Garbsen, Germany. The procedure consists in removing the soil air from the pores of the contaminated soil by pumping. In this way, an air flow field is created towards the pumps or *active* wells in the subsurface. Extracted air is replaced by clean one from the atmosphere, which enters the soil through an unsealed part of the surface, or so-called *passive* wells (open places to the atmosphere). Consequently, the volatile organic compounds in the gas-phase are removed from the soil. As the volatilisation process is very slow, the cleaning procedure may take over months or even years. On the other hand, according to the test fields, the time required to develop a steady state air flow in the subsurface is rather short (cf. [3]). The small volatilisation rate of the contaminants causes the under-pressure at the active wells to be small, compared to the absolute value of the atmospheric pressure. Hence, it offers no advantage to pump strongly in order to keep the amount of contaminants in the extracted gas-phase at a reasonable level.

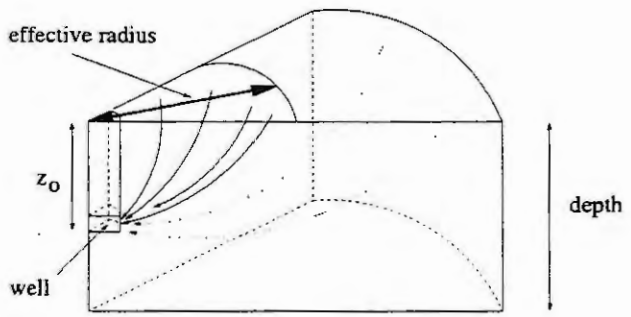


Figure 1: Unsealed surface

When the surface of a contaminated site is open to the atmosphere, the situation for a single extraction well looks as in Fig. 1. This is an inconvenient situation, because of the relatively small "effective" radius, causing only the vicinity of the well to be effectively cleaned. The value of the effective radius depends on the specific situation (depth of the domain, soil structure, heterogenity, soil surface, ...). For a detailed account on the effective radii of wells, we may refer to [3], [5] and [4]. In order to enlarge the effectively cleaned area, it is common practice to seal the soil surface with concrete, asphalt or iron plates. In this way the entrance of clean air from the atmosphere through the surface can be prohibited. The possibly leaky surface can be modelled by means of a leakage term in a Robin type BC.

Mathematical model

Let $\Omega \subset \mathbb{R}^3$ be a bounded open domain[1], with Lipschitz continuous boundary $\partial\Omega$, and containing n cylindrical active wells ($n \in \mathbb{N}$), with boundaries Γ_i^a, $(i = 1, \ldots, n)$, and m cylindrical passive wells ($m \in \mathbb{N}$), with boundaries Γ_i^p, $(i = 1, \ldots, m)$. We introduce the notation $\Gamma_A = \bigcup_{i=1}^n \Gamma_i^a$ and $\Gamma_P = \bigcup_{i=1}^m \Gamma_i^p$. The external boundary of Ω consists of two complementary parts Γ_D and Γ_N, where Dirichlet and Robin type BCs are prescribed, respectively. The situation is schematically shown in Fig. 2.

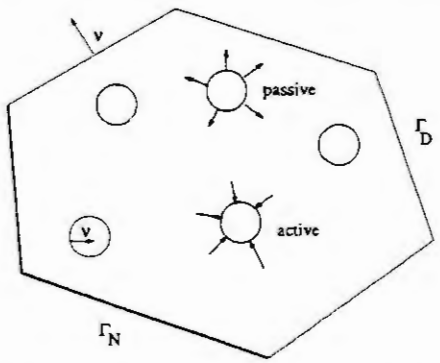


Figure 2: Top view of the domain with active and passive wells

We suppose that the gas in the pores of the soil obeys the ideal gas law. Denoting the pressure by p and introducing the variable $u = p^2$, the steady state air flow, induced by the active wells, is governed by a linear elliptic PDE for u, accompanied of appropriate BCs. This PDE reads

$$\nabla \cdot (-\boldsymbol{A}_{dif} \nabla u - \boldsymbol{a}_{con} u) + a_{sou} u = f \qquad \text{in } \Omega. \tag{1}$$

Here, $\boldsymbol{A}_{dif}(\boldsymbol{x})$ and $\boldsymbol{a}_{con}(\boldsymbol{x})$ correspond to the diffusive and convective part of the air flow at $\boldsymbol{x} \in \Omega$, respectively. Further, $f(\boldsymbol{x}) - a_{sou}(\boldsymbol{x})u$ represent the source term, supposed to be a linear function of u.

The most delicate point in the flow modelling is the choice of appropriate BCs at the active wells. Usually, the discharge (total flux through a well) can be measured, while the distribution of the flux itself along the boundary of an extraction well is unknown. Moreover, from the physical point of view we may assume that a constant, yet unknown under-pressure is induced at the boundary of each active well, by a given discharge. Thus we are led to a *nonlocal* Neumann BC, accompanied of a Dirichlet side condition at each boundary Γ_i^a, $(i = 1, \ldots, n)$, i.e.,

$$\int_{\Gamma_i^a} (-\boldsymbol{A}_{dif} \nabla u - \boldsymbol{a}_{con} u) \cdot \boldsymbol{\nu} = s_i \text{ (given)}, \tag{2}$$

$$u = c_i \text{ (unknown constant) on } \Gamma_i^a, \tag{3}$$

for a given set of discharges $s_i \in \mathbb{R}$, $(i = 1, \ldots, n)$. Here $\boldsymbol{\nu}$ stands for the outward unit normal vector.

In order to complete the BVP, BCs must be supplied at Γ_D, Γ_P and Γ_N. They may be taken to be

$$u = g_{Dir} \qquad \text{on } \Gamma_D \cup \Gamma_P, \tag{4}$$

$$(-\boldsymbol{A}_{dif} \nabla u - \boldsymbol{a}_{con} u) \cdot \boldsymbol{\nu} - g_{Rob} u = g_{Neu} \qquad \text{on } \Gamma_N. \tag{5}$$

The problem (1)–(5) can be interpreted as a problem of parameter identification (inverse problem): the unknown parameters c_i (squared pressures at Γ_i^a), entering (3), are sought in terms of the set of given (measured) total fluxes s_i through Γ_i^a, $(i = 1, \ldots, n)$.

The mathematical model (1)–(5) has been introduced in [1], where also the existence of an exact solution has been proved, under suitable regularity conditions on the data. However, in that paper, no method has been given for the construction of the solution. In fact, to avoid the difficulties connected with the nonlocal BCs (2), the authors have used Dirac functions to model the active wells as point sinks, i.e., the original problem (1)–(5) was replaced by an approximate one.

[1] We are considering a physical 3D-model, but the mathematical results obtained remain valid in a 2D-case as well. Models with reduced dimension have some practical relevance in particular situations (cf. [1]).

We deal with a new proof of the existence of a solution to the problem (1)–(5), which shows some important advantages. First, the proof also includes the uniqueness of the solution. Secondly, the proof is constructive: it provides a construction method for the solution, starting from the solution of well–posed auxiliary elliptic problems with *standard* (local) BCs. Finally, the method presented can be extended to the case where the active wells operate in groups, a strategy which is often used when designing soil venting facilities, viz when in each of the disjoint groups the wells work at the same specific under-pressure. The effectiveness of the method is illustrated by a numerical example.

Separately operating wells

Consider the function space on Ω,

$$V = \{\varphi \in W^{1,2}(\Omega) \mid \varphi = 0 \text{ on } \Gamma_D \cup \Gamma_P\},$$

where $W^{1,2}(\Omega)$ stands for the standard, first-order Sobolev space.

To simplify the notations, we denote by $G_i^a(u)$ the total flux through the boundary Γ_i^a of the i-th active well, i.e.,

$$G_i^a(u) = \int_{\Gamma_i^a} (-\boldsymbol{A}_{dif}\nabla u - \boldsymbol{a}_{con}u)\cdot\boldsymbol{\nu}, \quad i = 1,\ldots,n. \tag{6}$$

Next, we put

$$Au = \nabla\cdot(-\boldsymbol{A}_{dif}\nabla u - \boldsymbol{a}_{con}u) + a_{sou}u. \tag{7}$$

Moreover, we introduce the bilinear form $a(.,.)$ on $V\times V$ by

$$a(\varphi,\psi) = \int_\Omega (\boldsymbol{A}_{dif}\nabla\varphi + \boldsymbol{a}_{con}\varphi)\cdot\nabla\psi + \int_\Omega a_{sou}\varphi\psi + \int_{\Gamma_N} g_{Rob}\varphi\psi, \quad \forall\varphi,\psi\in V. \tag{8}$$

We assume that the bilinear form $a(.,.)$ fulfills the standard ellipticity property

$$\exists C_0>0:\ a(\varphi,\varphi) \geq C_0\,|\varphi|^2_{1,2,\Omega} \equiv C_0\int_\Omega |\nabla\varphi|^2, \quad \forall\varphi\in V. \tag{9}$$

Remark 1 We formulate a set of conditions on the data functions, appearing in (8), which ensure the V-ellipticity:

$$\begin{aligned}
\exists A_d>0:\ \int_\Omega \boldsymbol{A}_{dif}\nabla u\cdot\nabla u &\geq A_d\,|u|^2_{1,2,\Omega}, \quad \forall u\in V,\\
g_{Rob} &\geq 0 \ \text{ on } \Gamma_N,\\
a_{sou} &\geq A_s > \frac{\|\boldsymbol{a}_{con}\|^2_{0,\infty,\Omega}}{2\varepsilon A_d} \ \text{ in } \Omega, \ \text{ for some } \varepsilon\in(0,2).
\end{aligned}$$

Here, $\|.\|_{0,\infty,\Omega}$ stands for the usual $L_\infty(\Omega)$-norm.

By the Cauchy-Schwarz inequality in $L_2(\Omega)$ and a well-known algebraic inequality, we have

$$\begin{aligned}
\left|\int_\Omega \boldsymbol{a}_{con}u\cdot\nabla u\right| &\leq \|\boldsymbol{a}_{con}\|_{0,\infty,\Omega}\,\|u\|_{0,2,\Omega}\,|u|_{1,2,\Omega}\\
&\leq \frac{\|\boldsymbol{a}_{con}\|^2_{0,\infty,\Omega}}{2\varepsilon A_d}\,\|u\|^2_{0,2,\Omega} + \frac{A_d\varepsilon}{2}\,|u|^2_{1,2,\Omega},
\end{aligned}$$

where $\|.\|_{0,2,\Omega}$ denotes the $L_2(\Omega)$-norm.

On account of this estimate and the assumptions above, we arrive at

$$\begin{aligned}
\exists C>0:\ a(u,u) &\geq \left(A_d - \frac{A_d\varepsilon}{2}\right)|u|^2_{1,2,\Omega} + \left(A_s - \frac{\|\boldsymbol{a}_{con}\|^2_{0,\infty,\Omega}}{2\varepsilon A_d}\right)\|u\|^2_{0,2,\Omega}\\
&\geq C\,\|u\|^2_{1,2,\Omega}, \quad \forall u\in V.
\end{aligned}$$

□

First of all, we prove the uniqueness of the *weak* solution of the BVP considered.

Theorem 1 (uniqueness) *The BVP (1)-(5) has at most one solution.*

Proof: Suppose that u_1 and u_2 are solutions of (1)–(5), and denote $u = u_1 - u_2$. Then we have

$$\begin{aligned} Au &= 0 && \text{in } \Omega, && (10)\\ u &= \tilde{c}_i \text{ (constant)} && \text{on } \Gamma_i^a,\ i = 1,\dots,n, && (11)\\ G_i^a(u) &= 0 && i = 1,\dots,n, && (12) \end{aligned}$$

while moreover, u fulfills homogeneous Dirichlet and Robin BCs on $\Gamma_D \cup \Gamma_P$ and Γ_N, respectively. Multiplying both sides of (10) by u, integrating over Ω, applying Green's theorem and invoking (8) leads to

$$a(u,u) - \sum_{i=1}^{n} \int_{\Gamma_i^a} (\boldsymbol{A}_{dif} \nabla u + \boldsymbol{a}_{con} u) \cdot \nu u = 0,$$

or, on account of (11)–(12), $a(u,u) = 0$. As $u \in V$, we infer from the ellipticity condition (9) that $u = 0$ a.e. in Ω, i.e., $u_1 = u_2$. □

To prepare the *constructive* proof of the existence theorem, we consider the following auxiliary problems, related to (1)–(5):

$$\left.\begin{aligned} \nabla \cdot (-\boldsymbol{A}_{dif} \nabla v - \boldsymbol{a}_{con} v) + a_{sou} v &= f && \text{in} && \Omega\\ v &= g_{Dir} && \text{on} && \Gamma_D \cup \Gamma_P\\ (-\boldsymbol{A}_{dif} \nabla v - \boldsymbol{a}_{con} v) \cdot \nu - g_{Rob} v &= g_{Neu} && \text{on} && \Gamma_N\\ v &= 0 && \text{on} && \Gamma_A, \end{aligned}\right\} \qquad (13)$$

and for, $i = 1,\dots,n$,

$$\left.\begin{aligned} \nabla \cdot (-\boldsymbol{A}_{dif} \nabla z_i - \boldsymbol{a}_{con} z_i) + a_{sou} z_i &= 0 && \text{in} && \Omega\\ z_i &= 0 && \text{on} && \Gamma_D \cup \Gamma_P\\ (-\boldsymbol{A}_{dif} \nabla z_i - \boldsymbol{a}_{con} z_i) \cdot \nu - g_{Rob} z_i &= 0 && \text{on} && \Gamma_N\\ z_i &= 1 && \text{on} && \Gamma_i^a\\ z_i &= 0 && \text{on} && \Gamma_A \setminus \Gamma_i^a. \end{aligned}\right\} \qquad (14)$$

From the theory of linear elliptic equations (cf., e.g., [2]), we know that each of these problems has a unique weak solution under classical regularity conditions for the data.

Now, for any $\alpha = (\alpha_1,\dots,\alpha_n) \in \mathbb{R}^n$ we define

$$u_\alpha := v + \sum_{i=1}^{n} \alpha_i z_i. \qquad (15)$$

By linear superposition, the function u_α is seen to obey

$$\left.\begin{aligned} \nabla \cdot (-\boldsymbol{A}_{dif} \nabla u_\alpha - \boldsymbol{a}_{con} u_\alpha) + a_{sou} u_\alpha &= f && \text{in} && \Omega\\ u_\alpha &= g_{Dir} && \text{on} && \Gamma_D \cup \Gamma_P\\ (-\boldsymbol{A}_{dif} \nabla u_\alpha - \boldsymbol{a}_{con} u_\alpha) \cdot \nu - g_{Rob} u_\alpha &= g_{Neu} && \text{on} && \Gamma_N\\ u_\alpha &= \alpha_i && \text{on} && \Gamma_i^a,\ i = 1,\dots,n. \end{aligned}\right\} \qquad (16)$$

The flux of u_α through the boundary of the j-th active well takes the form

$$G_j^a(u_\alpha) = G_j^a(v) + \sum_{i=1}^{n} \alpha_i G_j^a(z_i).$$

In order to solve the original problem (1)–(5), we are looking for an n-tuple α, for which

$$G_j^a(u_\alpha) = s_j, \qquad j = 1,\dots,n.$$

This leads to a linear algebraic system

$$\begin{pmatrix} G_1^a(z_1) & \dots & G_1^a(z_n) \\ \vdots & \vdots & \vdots \\ G_n^a(z_1) & \dots & G_n^a(z_n) \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{pmatrix} + \begin{pmatrix} G_1^a(v) \\ \vdots \\ G_n^a(v) \end{pmatrix} = \begin{pmatrix} s_1 \\ \vdots \\ s_n \end{pmatrix}. \qquad (17)$$

Now, we are in a position to prove the existence of a solution of the BVP (1)–(5).

Theorem 2 *There exists a solution of the BVP (1)–(5).*

Proof: According to the considerations above, it is sufficient to show the regularity of the matrix $\boldsymbol{G} = (G_i^a(z_j))_{i,j}$ entering (17). To this end, suppose that $\boldsymbol{G}$ is singular. Then, one of the columns of $\boldsymbol{G}$ is a linear combination of the other columns. Without loss of generality we may assume that, for some constants $\lambda_1, \ldots, \lambda_{n-1}$,

$$G_j^a\left(z_n - \sum_{i=1}^{n-1} \lambda_i z_i\right) = 0, \qquad \text{for } j = 1, \ldots, n.$$

From this relation and from (14), the function $w = z_n - \sum_{i=1}^{n-1} \lambda_i z_i$ is seen to solve the problem

$$\left.\begin{array}{rll} \nabla \cdot (-\boldsymbol{A}_{dif} \nabla w - \boldsymbol{a}_{con} w) + a_{sou} w = 0 & \text{in} & \Omega \\ w = 0 & \text{on} & \Gamma_D \cup \Gamma_P \\ (-\boldsymbol{A}_{dif} \nabla w - \boldsymbol{a}_{con} w) \cdot \boldsymbol{\nu} - g_{Rob} w = 0 & \text{on} & \Gamma_N \\ w = const & \text{on} & \Gamma_i^a \\ G_i^a(w) = 0 & \text{on} & \Gamma_i^a,\ i = 1, \ldots, n. \end{array}\right\}$$

Clearly, $w = 0$ is a solution of this problem. By Theorem 1 this zero solution is the only solution. We conclude that

$$z_n = \sum_{i=1}^{n-1} \lambda_i z_i. \tag{18}$$

Recalling the fact that, by (14), $z_1, \ldots, z_{n-1}$ vanish on Γ_n^a, we get $z_n = 0$ on Γ_n^a. This is contradictory with the BC $z_n = 1$ on Γ_n^a, see also (14). Hence, the matrix $\boldsymbol{G}$ is regular. Consequently, the algebraic system (17) has a unique solution $\boldsymbol{\alpha}$. The unique solution of the BVP (1)–(5) is then defined by the relation (15). □

We emphasize the constructive character of the existence proof. The desired solution of the problem (1)–(5) has been expressed in terms of solutions of some auxiliary problems with standard BCs. Moreover, the values $\alpha_1, \ldots, \alpha_n$ are nothing else than the, a priori unknown, values of the squared pressure at the respective boundaries $\Gamma_1^a, \ldots, \Gamma_n^a$ of the active wells.

In practice, the auxiliary problems (13) and (14) are solved by suitable approximation methods, such as finite element methods (FEMs).

Wells operating in groups

As mentioned above, soil venting is a slow process, which can take over months or years. Due to the small volatilisation rates of the organic contaminants, the under-pressures at all active wells should be low. On the other hand, the engines for extraction wells are nowadays very effective and powerful. Hence, it is useful to design the soil venting facilities in such a way that the active wells are splitted into disjoint groups, each group being connected with a single engine, i.e., the under-pressures at all extraction wells inside one group are equal.

We suppose to have k groups ($k < n$). Let $W_j \subset \{1, \ldots, n\}$ denote the set of indices of the active wells belonging to the j-th group. The discharge conditions (2)–(3) have to be modified into the form

$$\left.\begin{array}{rll} u = \alpha_j = const \ \text{(unknown)} & \text{on} & \bigcup_{i \in W_j} \Gamma_i^a \\ \sum_{i \in W_j} G_i^a(u) = s_j \in \mathbb{R} & & \end{array}\right. \tag{19}$$

for $j \in \{1, \ldots, k\}$.

Recall that all simultaneously working wells inside one group operate at the same under-pressure, i.e., this group has only one degree of freedom. Thus, for $j = 1, \ldots, k$, we consider the function

$$y_j = \sum_{i \in W_j} z_i, \tag{20}$$

where z_i is the unique solution of problem (14). For any $\alpha = (\alpha_1, \ldots, \alpha_k) \in \mathbb{R}^k$, we introduce the function

$$u_\alpha = v + \sum_{j=1}^{k} \alpha_j y_j. \tag{21}$$

Here, again, v denotes the unique solution of the BVP (13). By the superposition principle, the function u_α is now seen to obey

$$\left.\begin{array}{rll} \nabla \cdot (-A_{dif} \nabla u_\alpha - a_{con} u_\alpha) + a_{sou} u_\alpha = f & \text{in} & \Omega \\ u_\alpha = g_{Dir} & \text{on} & \Gamma_D \cup \Gamma_P \\ (-A_{dif} \nabla u_\alpha - a_{con} u_\alpha) \cdot \nu - g_{Rob} u_\alpha = g_{Neu} & \text{on} & \Gamma_N \\ u_\alpha = \alpha_j & \text{on} & \bigcup_{i \in W_j} \Gamma_i^a,\ j = 1, \ldots, k. \end{array}\right\}$$

Analogously as before, the flux of u_α through $\bigcup_{i \in W_j} \Gamma_i^a$ reads

$$G^a_{W_j}(u_\alpha) = G^a_{W_j}(v) + \sum_{i=1}^{k} \alpha_i G^a_{W_j}(y_i), \tag{22}$$

where

$$G^a_{W_j}(w) = \sum_{i \in W_j} G^a_i(w).$$

To solve the original, modified problem, we are looking for an $\alpha \in \mathbb{R}^k$, satisfying

$$G^a_{W_j}(u_\alpha) = s_j, \qquad j = 1, \ldots, k.$$

Hence, we have to solve the linear algebraic system

$$\begin{pmatrix} G^a_{W_1}(y_1) & \ldots & G^a_{W_1}(y_k) \\ \vdots & \vdots & \vdots \\ G^a_{W_k}(y_1) & \ldots & G^a_{W_k}(y_k) \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_k \end{pmatrix} + \begin{pmatrix} G^a_{W_1}(v) \\ \vdots \\ G^a_{W_k}(v) \end{pmatrix} = \begin{pmatrix} s_1 \\ \vdots \\ s_k \end{pmatrix}. \tag{23}$$

In a similar way as for separately operating wells we may show that the matrix $(G_{W_i}(y_j))_{i,j}$ is regular. The solution α of the system (23) determines the pressure at the boundaries of the extraction wells in each of the k groups, α_l being the pressure corresponding to the l-th group ($l = 1, \ldots, k$). Then, the solution u_α of the modified problem is the function given by (21).

A numerical example

In this section we present a numerical example, representing a model situation in soil venting. We consider a cylindrical domain in $\mathbb{R}^3$ with a vertical axis. We assume that this domain is insulated at all sides (at the bottom by an impervious layer, at the top by an iron plate or concrete, and at the lateral surface by an artificially constructed barrier). Fresh air can enter the domain through one passive well which is open to the atmosphere.

When we suppose the geological soil layers to be horizontal and the extraction wells to pump air along the whole of their vertical length, a horizontal air flow field will be created. Vertical integration then yields a reduced 2D-model (cf. [1]). An example of such a reduced 2D-situation is represented in Fig. 3, where three active wells, with respective boundaries Γ_1, Γ_2 and Γ_3, and one passive well with boundary Γ_P are considered. The geometrical data are as follows. The domain is centered at the origin and has radius 2. All wells (active and passive) have radius 0.2. The passive well is also centered at the origin, while the midpoints of the active wells are respectively given by $(1.5, 0)$, $(-1.2, 0.5)$ and $(0, -1.0)$.

We consider the following model problem:

$$\left.\begin{array}{rll} -\Delta u = 0 & \text{in} & \Omega \\ u = 0 & \text{on} & \Gamma_P \\ -\nabla u \cdot \nu = 0 & \text{on} & \Gamma_N \\ u = c_i \text{ (unknown)} & \text{on} & \Gamma_i,\ i = 1, 2, 3, \\ \int_{\Gamma_i} -\nabla u \cdot \nu = s_i & & i = 1, 2, 3, \end{array}\right\}$$

for $s_1 = 0.2$, $s_2 = 0.3$ and $s_3 = 0.25$.

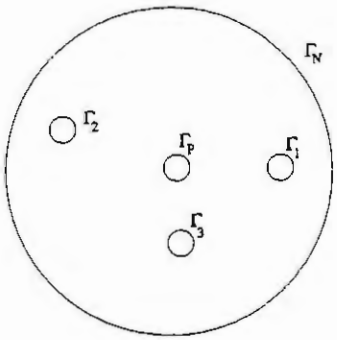


Figure 3: Situation

This problem is solved by the method outlined in Section 3. As the auxiliary problem (13) now is completely homogeneous, we have $v = 0$. To solve the three auxiliary problems of the type (14), we use a FEM, with piecewise linear polynomials on a triangular mesh. The unstructured triangulation of the domain consists of 21888 triangles. The isolines for z_i, $i = 1, 2, 3$, are shown in Fig. 4. The corresponding

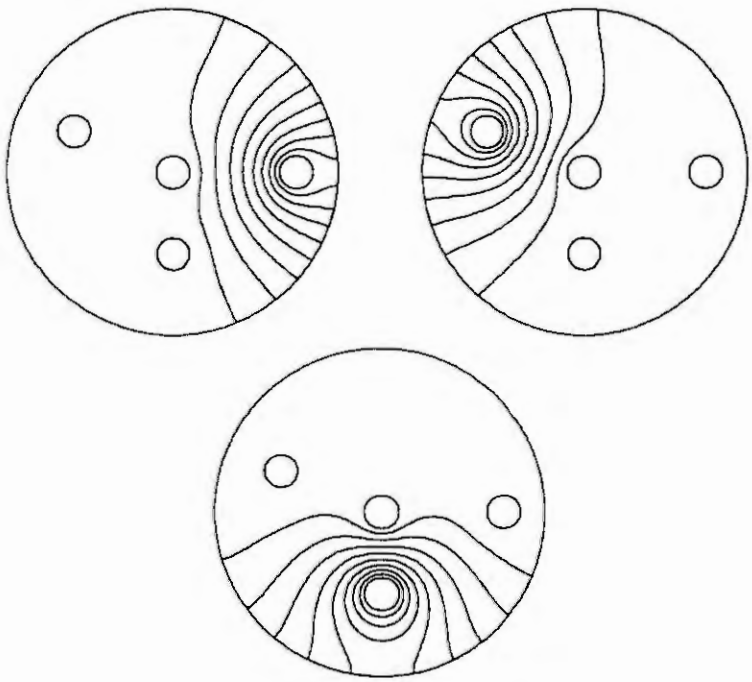


Figure 4: Isolines for z_1, z_2 and z_3

algebraic system of the type (17) is found to be

$$\begin{pmatrix} -1.77707 & 0.0220422 & 0.274207 \\ 0.0223199 & -2.14177 & 0.166005 \\ 0.287506 & 0.16705 & -2.63833 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{pmatrix} = \begin{pmatrix} 0.2 \\ 0.3 \\ 0.25 \end{pmatrix}.$$

Its solution reads

$$\alpha_1 = -0.1327387057, \quad \alpha_2 = -0.1506593337, \quad \alpha_3 = -0.1187610405.$$

The equipotential curves for the corresponding solution u_α, (15), are shown in Fig. 5.

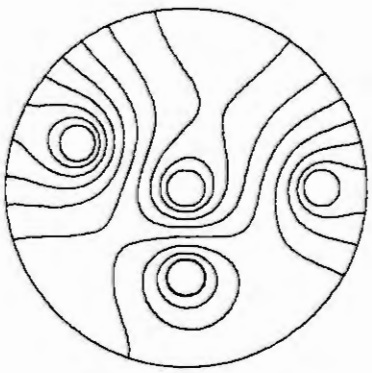


Figure 5: Isolines for u_α

To check the reliability of the method, we consider the following elliptic problem, with standard Dirichlet BCs on Γ_i, given by the values obtained for α_i, $(i = 1, 2, 3)$:

$$\left.\begin{array}{rll} -\Delta u_\alpha &= 0 & \text{in } \Omega \\ u_\alpha &= 0 & \text{on } \Gamma_P \\ -\nabla u_\alpha \cdot \nu &= 0 & \text{on } \Gamma_N \\ u_\alpha &= -0.1327387057 & \text{on } \Gamma_1 \\ u_\alpha &= -0.1506593337 & \text{on } \Gamma_2 \\ u_\alpha &= -0.1187610405 & \text{on } \Gamma_3. \end{array}\right\}$$

When solving this well–posed problem by the FEM mentioned above, the (approximate) values of the corresponding fluxes through the boundaries of the active wells are found to be

$$G_1(u_\alpha) = 0.2, \quad G_2(u_\alpha) = 0.299999, \quad G_3(u_\alpha) = 0.25.$$

These calculated fluxes coincide almost exactly with the prescribed values s_1, s_2 and s_3, respectively.

Summary

The method above allows to recover the pressure values at active wells, either operating separately or in groups, from their measured discharges. This method offers the advantage to be constructive, i.e., the desired solution is found in terms of the solutions of some auxiliary problems of a simpler type.

Acknowledgement. One of the authors (MS) was financially supported by the VEO-project no. 011 VO 697. The authors thank R. Van Keer, coordinator of this project, for stimulating discussions and for his critical reading of the text.

References

1. Gerke, H., Hornung, U., Kelanemer, Y., Slodička, M. and Schumacher, S., Optimal Control of Soil Venting: Mathematical Modeling and Applications. Birkhäuser, Basel, 1999.
2. Gilbarg, D. and Trudinger, N.S., Elliptic Partial Differential Equations of Second Order. Springer, Berlin-Heidelberg, 1983.
3. Hiller, D.H., Performance charasteristics of vapor extraction systems operated in Europe. In: Proc. Symposium on Soil Venting, Houston, Texas, 1991, 193–202. [Environmental Protection Agency EPA/600/R-92/174].
4. Schumacher, S. and Slodička, M., Effective radius and underpressure of an air extraction well in a heterogeneous porous medium. Transport in Porous Media, 29 (1997), 323–340.
5. Wilson, D.J., Modeling of insitu techniques for treatment of contaminated soils: Soil vapor extraction, sparging, and bioventing. Technomic Publishing Company Inc., Lancester-Basel, 1995.

A STUDY OF PARAMETRIC MODELS APPLIED TO IN-SERVICE LIFE PREDICTION OF DRY VACUUM PUMP

Arihiro Ishida, Satoshi Konishi, Toshiro Sato, and Kiyohito Yamasawa
Dept. of Electrical & Electronic Engng., Shinshu University,
Wakasato 4-17-1, Nagano, 380-8553 Japan

Abstract. We have studied the possibility of prediction for the in-service life (abbreviated as ISL, hereinafter) of dry pumps from the unstable non-stationary motor current which is observed before over-load pump stopping. A model describing the data as-measured (*i.e.* non-stationary model) and stationary models converted from the non-stationary one have been evaluated. The data were obtained from an accelerated test performed under TEOS process. It has been found that the stationary models can predict ISL in more stable manner than the non-stationary ones. According to the analytical result of each model, it is concluded that, in the view of practice of ISL prediction, the importance is at the selection of models that leads to stable prediction resulting in high certainty of ISL prediction.

Introduction

Dry vacuum pumps have been widely used for various semiconductor fabrication processes [1] due to the easy maintainability and long-term in-service life (abbreviated as ISL). The actual ISL upon pump seizure is not, however, easily predicted from the operational status. Recently, a simple system for the diagnosis of the pump ISL was developed [2]. The systematic evaluation of stochastic models applied to pump ISL prediction has not done. In the paper, the models are evaluated by MATLAB in use for system identification [3], [4].

Acceleration test data

Fig. 1 shows a set of acceleration test data for pump current used for the study. The data were obtained by TEOS (Tetraethoxysilane) CVD which is one of the most harsh process for dry pumps. The whole life of the pump consists of stationary state (SS) and final state (FS). The SS term features white noise characteristics (therefore stationary) but the FS one shows a trend due to a finite time lag (non-stationary) both seen in auto-covariance analysis (abbreviated as ACA). The existence of a trend in FS leads to the possibility of prediction of ISL, provided the models have to reasonably behave in both SS and FS terms.

Stochastic models

We have selected stochastic models from linear parametric models as; ARMA (Auto-Regressive Moving Average), ARMAX (Auto-Regressive Moving Average eXogenous), ARIMA (Auto-Regressive Integrated Moving Average) and PX (Pseudo eXogenous) models. The PX model has been newly devised in this study.

A well-known method to evaluate the sationariness is to take auto-covariance analysis (ACA). Fig. 2 shows the results of ACA. The differentiated observation by the estimated bias in PX model shows the similar stationariness to that of the ARIMA. The other models have no enhancement of stationariness, since their ACA are given by the signals as observed.

The difference between the estimate and the signal observed in the next step should be given in white noise model. The difference of the ACA for a set ARMA, ARIMA, ARMAX and PX models could not be observed significantly. It verifies the signals are properly identified.

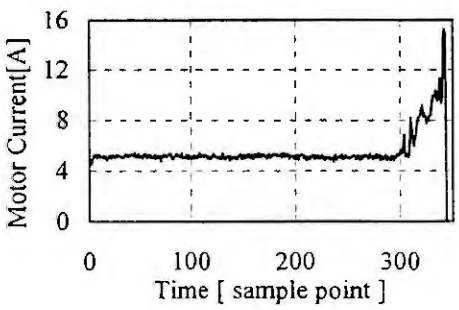
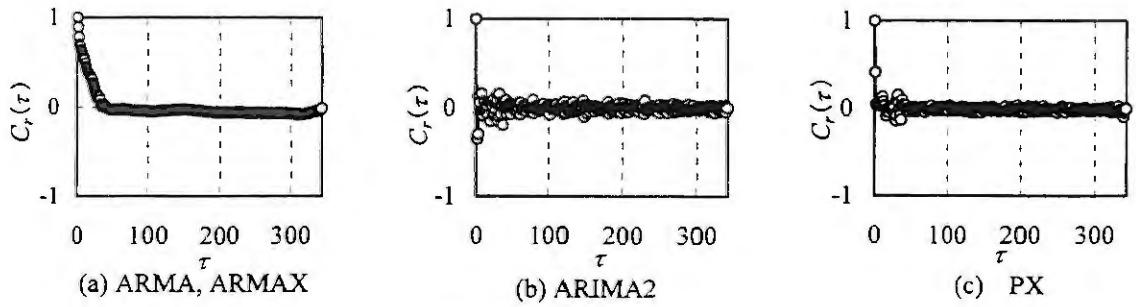
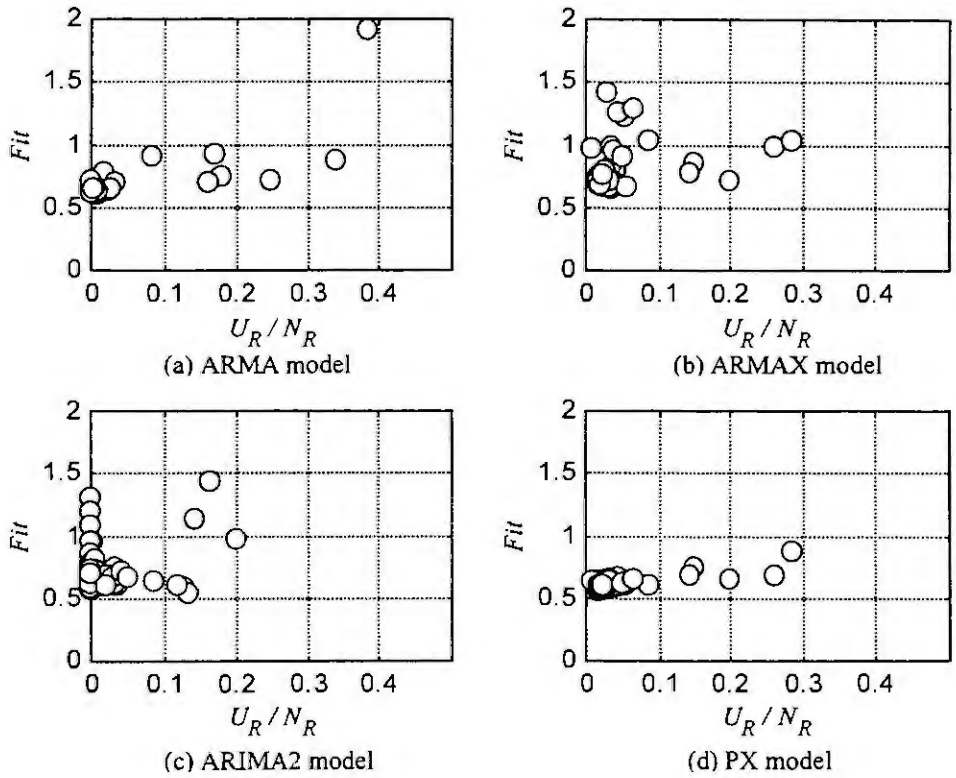


Fig.1. Acceleration test data



(a) ARMA, ARMAX (b) ARIMA2 (c) PX

Fig. 2. ACA results



(a) ARMA model (b) ARMAX model

(c) ARIMA2 model (d) PX model

Fig.3. The connection between stability and five-step prediction error

Model stability and Immunity from the parametric dimension

The auto-regressive part of signal models has an important rule such that the characteristic polynomial equations have all solutions in a unit circle. Otherwise, the predicted values may unrealistically diverse from the current estimate given at the time of identification of the model. We took an averaged stability index as U_R/N_R calculated from the solutions of the characteristic equation. For the degree of diversion in prediction, *Fit* (*r.m.s.q.* of summation of prediction error) was evaluated. Fig. 3 shows *Fit vs.* U_R/N_R. It can be understood that the more the instability the more the error of prediction.

For a diverting signal, a proper selection of the parameter dimension may give a well fitting estimate to the observed signal. AIC [5] gives a good determination of the dimension. However, for the versatility of models such that they have to be unchanged for SS and FS does not allow to have different dimension for each term. The key is to find a robust model irrespective of the parameter dimensions. Fig. 4 shows the *Fit* for each model *vs.* AR (Auto-Regressive) parameters and MA (Moving Average) parameters. ARIMA2 shows remarkable immunity against the parameter dimension.

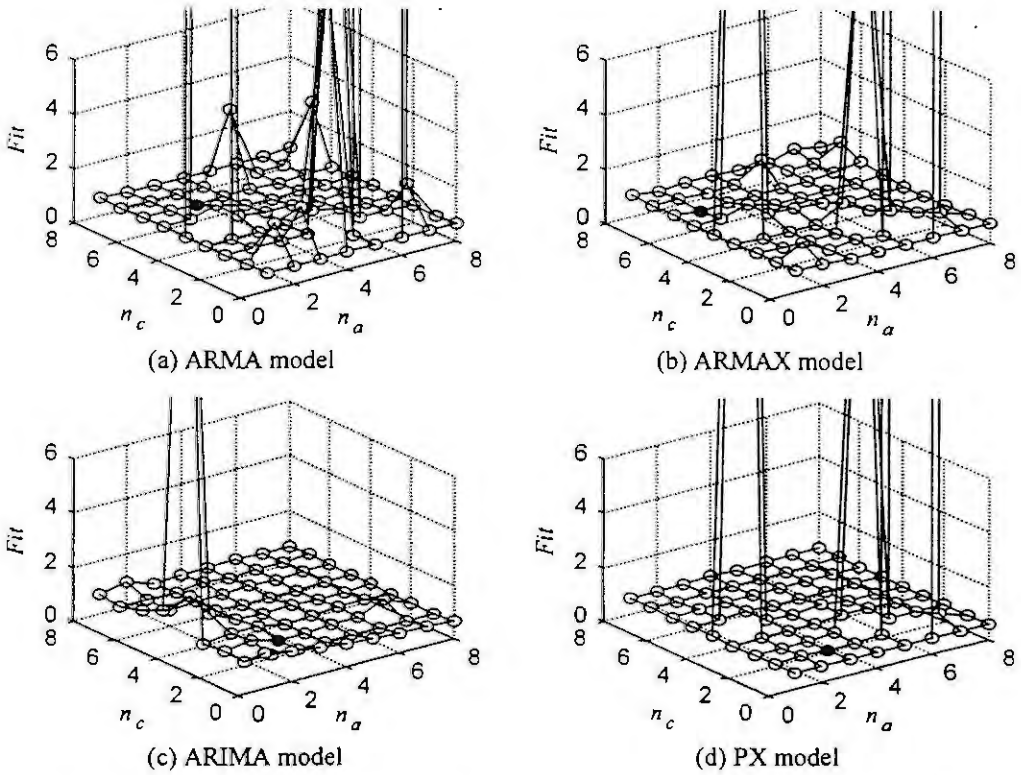


Fig. 4. The connection between five-step prediction error and parameter dimension

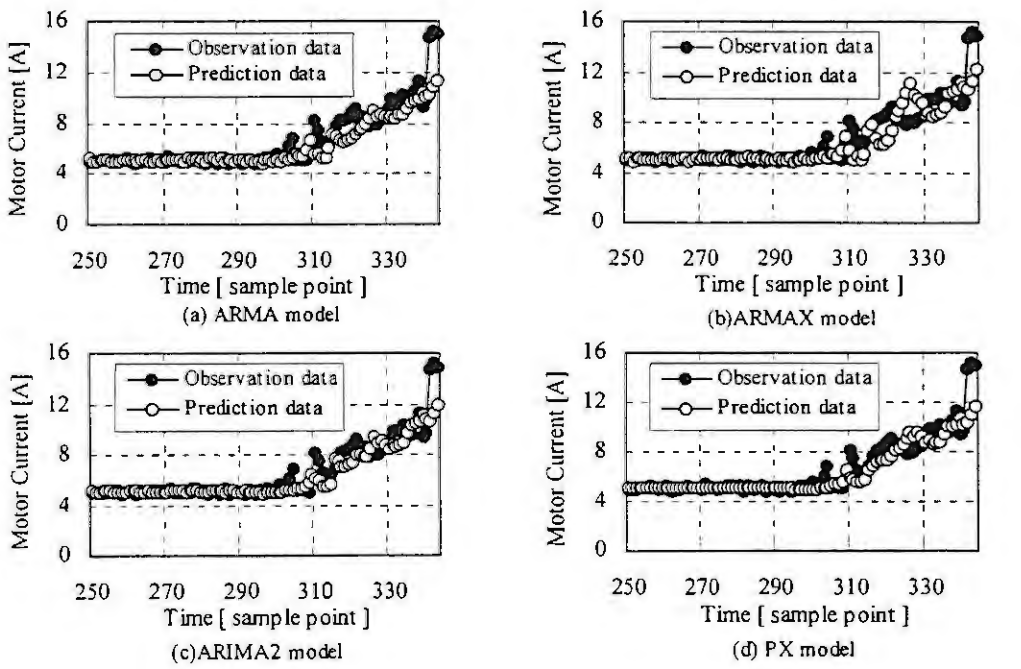


Fig.5. Five-step prediction of each model

The prediction with the actual observation

The stationariness and parameter immunity are important aspects for the sureness of the prediction, however, the more direct and overall evaluation of the prediction is to examine the difference between the predicted data and actual ones. Fig. 5 shows the comparison of these data for five-step prediction. It can be seen that ARIMA2 and PX are superior to the other models.

Table 1. Characteristic of each model

	Instability model		Stability model	
	ARMA	**ARMAX**	**ARIMA2**	**PX**
1) Stationariness evaluation by auto-covariance	Unstable Fig.2 (a)	Unstable Fig.2 (a)	Stable Fig.2 (b)	Stable Fig.2 (c)
2) Whiteness of one-step prediction	Sufficient	Sufficient	Sufficient	Sufficient
3)Model stability	Acceptable Fig.3 (a)	Sufficient Fig.3 (b)	Sufficient Fig.3 (c)	Sufficient Fig.3 (d)
4) Immunity from the parameter dimension	Acceptable Fig.4 (a)	Sufficient Fig.4 (b)	Sufficient Fig.4 (c)	Sufficient Fig.4 (d)
5) Correspondency of the prediction with the actual observation	Acceptable Fig.5 (a)	Acceptable Fig.5 (b)	Sufficient Fig.5 (c)	Sufficient Fig.5 (d)

Conclusion

We have evaluated several stochastic models for the purpose of "good" ISL prediction of a dry vacuum pump. The importance of the evaluation is from the fact that there are not many models that can give a reasonable prediction for the whole term of pump operation due to its non-stationary signal behavior. The result is summarized in table 1. The prediction using good-resulted models can serve for real-time ISL determination which may realize are reliable facility maintenance of semiconductor fabrication.

References

1) H. Wycliffe, "Mechanical high-vacuum pumps with an oil-free swept volume", *J. Vac., Sci. Tech.* A5 (4), Jul/Aug 1987
2) S.Konishi and K.Yamasawa, "Diagnostic system to determine the in-service life of dry vacuum pumps", IEE Proc. Science, Measurement and Technology, 146(6), Nov,1999 (to be published)
3) G. C. Goodwin and K. S. Sin, Adaptive Filtering Prediction and Control, PRENTICE-HALL, INC, 1984
4) Lennart Ljung, System Identification: Theory for the user, PRENTICE-HALL, INC, 1987
5) H. Akaike, "A New Look on the Statistical Model Identification", *IEEE Trans.* AC-19, pp. 716-723 (1974)

FORECAST OF THE SLAGGING TENDENCY OF PULVERIZED COAL FIRED FURNACES BY SIMULATION OF MINERAL MATTER TRANSFORMATION IN THREE-DIMENSIONAL MULTI-PHASE FLOW FIELD

O. Bozic, R. Leithner, H. Müller
Institut für Wärme- und Brennstofftechnik
Technical University of Braunschweig
Franz – Liszt – Str. 35, D-38106 Braunschweig, Germany

Abstract. During the combustion of pulverised coal in the furnaces of power plant slagging can occur, which leads under some conditions to shut down for cleaning. For this reason the forecast of the slagging, still in the project and design phase of a plant, is of particular importance. It is also of the same importance for plants, which intend to buy cheap coal on a world wide basis, to forecast if the coal offered leads to slagging and accompaning problems or not. The article shows a new way to forecast the slagging tendency using computer simulations. At the Institut für Wärme- und Brennstofftechnik (IWBT) of the TU Brunswick the CFD program package FLOREAN was combined with the postprocessor program TRAMIC (TRAnsformation of the MIneral Components) to calculate the slagging tendency for determined furnaces and coals. As example, simulation of a combustion and slagging process in a furnace of the IFRF IJmuiden with jet burner and with high-volatile bituminous coal are shown.

1. Introduction

The existing methods for slagging tendency estimation of furnaces of solid fuel fired boilers, which use characteristic ratios and proportions (till today more than 70 ratios are in use) are inadequate. One of the reasons is, that those methods use oxidic ash analysis. Accordingly, real conditions in furnaces (temperature, gas atmosphere), which determine the way and speed of the mineral matter transformation are not considered by those methods. The second reason is that furnace dimensions and burner design is not included at all. Using features that modern computer technique offers, a more promising procedure for slagging tendency forecasting was established. Starting with the coal minerals at the burners the mineral matter transformation along the particle trajectories by heterogeneous reactions and reactions in solid state is calculated, taking into consideration the local temperatures and concentrations in the boiler furnace. The modelling procedure is performed in three steps, described in this paper. In addition first results are presented.

2. Modelling Procedure

2.1 Calculation of Continuous Phase

In the first step the CFD program package FLOREAN calculates velocity-, gas temperature- and concentration-fields for the combustion products (flue gas) within the given furnace. The gas representing the continuous phase is described with an Eulerian approach, which implies conservative character of the general transport equations

$$\frac{\partial}{\partial t}(\rho\overline{\Phi}) = -\frac{\partial}{\partial x_i}(\rho\, u_i\overline{\Phi}) + \frac{\partial}{\partial x_i}\left(\Gamma_\Phi \frac{\partial\overline{\Phi}}{\partial x_i}\right) + S_\Phi \qquad (1)$$

Equation (1) represents the partial differential equations for the transport of mass, momentum, energy, turbulence and species. The meaning of the quantities $\overline{\Phi}$, Γ_Φ and S_Φ in the different transport equations is given in the table 1.

Transport (balance) equations were transformed into systems of algebraic equations using the Finite-Volume-Method, the UPSTREAM algorithm, which provides stability and were than solved by TDM-Algorithm. An orthogonal three-dimensional grid was applied. For pressure correction (mass balance) SIMPLE algorithm was used. The influence of the turbulence on the flow field, the energy and mass transportation was taken into consideration with the k–ε turbulence-model. The Six-Flux-Model with variable integration angles was used for the simulation of the radiative heat transfer. FLOREAN can calculate the concentration of coal and of other solid particles in two different ways. Using Eulerian approach solid particles are considered as heavy gas component.

This simplification is allowed, because in pulverised coal fired furnaces the particle concentration is very low and the particles are small, so that they follow the gas flow with negligible slip velocity. In the simulated example only one particle size with mass-weighted mean diameter of the particle size distribution was used. An alternative way for calculation of processes in the solid phase is explained in chapter 2.2.

Table 1: System of balance equations

Balance	Quantity $\overline{\Phi}$	Exchange Coefficient Γ_Φ	Sink/ Source Term S_Φ
Mass	1	0	0
Momentum	$\overline{u_i}$	μ_{eff}	$-\frac{\partial p}{\partial x_i} + g_i \rho_G$
Turbulent Kinetic Energy	k	μ_{eff} / σ_k	$G_k - \rho\varepsilon$
Turbulent Dissipation Rate	ε	$\mu_{eff} / \sigma_\varepsilon$	$\frac{\varepsilon}{k}\left(C_1 G_k - C_2 \rho\varepsilon\right)$
Energy	h	μ_{eff} / σ_h	$S_{rad} + S_{chem}$
Radiation: 6-Flux-Model	$\frac{\partial}{\partial x_i}\left(\frac{1}{K_a}\frac{\partial}{\partial x_i} b_{ij} B_i\right) = +K_a B_i - \frac{K_a}{\pi}\sigma T^4$		
Mixture Fraction	$\tilde{f}$	μ_{eff} / σ_h	$S_f = 0.0$
Components (O_2, CO_2, H2O, CO, C_xH_y)	Y_n	μ_{eff} / σ_Y	S_Y
	Y_n	μ_{eff} / σ_Y	S_Y
	.....	.....	.....
Pollutants (HCN,NH_3,NO)	Y_n	μ_{eff} / σ_Y	S_Y

with $G_k = \left[\mu_t\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) - \frac{2}{3}\rho k \delta_{ij}\right]\frac{\partial \overline{u}_i}{\partial x_j}$

2.2 Calculation of the Disperse Phase

In the second step the trajectories of a great number of coal particles inside the given flow field (without influence on the flow field) are calculated till the collision of the particles with the furnace walls or till the outlet of the furnace (Lagrangian approach with one-way-coupling). In FLOREAN two-way coupling (then the gas phase does not contain solid particles) is also possible, but results in very high computing time.

For the calculation of particle trajectories, only mass, drag, gravitational and buoyancy forces were considered while all other forces were neglected. The influence of the fluid turbulence on the trajectories was considered by a stochastic approach. Along the trajectory of each particle the combustion process, including drying of row coal (McIntosh model), pyrolysis (one-step-model), volatile combustion (Eddy-Dissipation-Concept), char burn-out (Field model) is calculated so that at the end nearly only ash remains. During the particle combustion along the trajectory the particle size and density (Shadow-method) and the particle temperature are also calculated. A more detailed description of all models is given in [2]. The particle temperature was calculated from the energy balance of the particle taking into consideration the heat fluxes due to convection, radiation, evaporation and combustion (source term through char burn-out and sink term through particle mass decrease). The differential equation for particle velocities, the particle position, the particle components water, pure coal, coke and particle temperature were solved with the Euler-Cauchy-Method. It was assumed that the mass of uncombustibles (ash) in coal particle remains constant and does not take part in any chemical reaction (assumption is valid for this calculations step). A simple sticking criterion (temperature) decides if one particle sticks to the wall after collision or is reflected.

2.3 Mineral Transformation

All coal particles and mineral particles are assumed to be spheres, in certain cases with concentric nucleus. Minerals can be transformed by numerous different physical and chemical processes: chemical reactions, diffusion, melting, dissolution, amorphous solidification, crystallisation, glass-formation out of crystal and chemical decomposition of the crystal-phase. Miscellaneous processes appear together or independent so that the number of the combinations becomes very large. Models which describe those processes preferably use simple algebraic equations and analytically integrated differential equations (when is possible), because of saving computing time. For all processes a very important question is to find the change of :

$$\textit{volume fraction}\quad X_i = \frac{V_i(t)}{V_O} \qquad \text{or} \qquad \textit{mass fraction}\quad Y_i = \frac{m_i(t)}{m_o} = \frac{\rho_i(T,\varepsilon_{por})}{\rho_o} X_i \qquad (2)$$

as function of time. In general transformations can be expressed by the following equations:

$$\sum_{E=1}^{n} Y_E = \text{TRANSFORMATION} \Rightarrow \sum_{P=1}^{m} Y_P \qquad (3)$$

i. e. according to the balance of matter educts and products have to correspond. A general transformation model independent from the type of the process can be given as :

$$\frac{dX}{dt} = \mathbf{K}\, f(X) \qquad \mathbf{K}\, t = g(X) \qquad g(X) = \int_O^X \left[\frac{1}{f(X)}\right] dX \qquad \mathbf{K} = \mathbf{k_0}\, \mathbf{exp}\left(-\frac{E_A}{\Re T}\right) \quad (4, 5, 6, 7)$$

The inverse integral function $X = G^I(K\ t)$ describes time-dependent kinetics of these heterogeneous process, which generally cannot be described successfully with thermodynamic modelling (based on minimising of the free enthalpy).

At the Technical University of Braunschweig [6] series of real transformations were investigated and grouped as *simple processes* [nucleation and nuclei's growth (**An**), formal kinetic n^{th} step (**Fn**), **n**-dimensional diffusion (**Dn**), phase-boundary-controlled-reactions (**Rn**)] or *complex processes* e.g. diffusion and simultaneous chemical reactions (Shrinking-Core-Model, Grain-Shrinking-Core-Model and others). Total different processes can be described with the same equations (in the differential or integral form). For example, with **R3** model a chemical reaction in the mineral particle with nonporous core can be described. In that case, diffusion resistance can be neglected at all. The same model can describe evaporation, sublimation or dissolution, of course with different functions for the transformation constants K. For description of heterogeneous chemical reaction, the phenomenological model **Fn** can have a useful role, the form of which is based on the analogy with equations for chemical reactions in gaseous states. The model **An** (Johnson-Mehl-Avrami-Kolmogorov-Type) can describe different cases of crystallisation, recrystallisation and decomposition. With the model D1 coalescence of two mineral phases in one particle by solid state diffusion can be described. The models D3 – D5 (s. table 2) describe solid state diffusion of binary mineral powder mixture compressed in a particle.

In the cases where gas diffusion, Knudsen diffusion or some other art of diffusion is coupled with chemical reaction f. e. Shrinking Core Model (SCM) can be applied. The important feature of this model (s. table 3) is that the mineral core (educts) is nonporous, while the spherical envelope (products) can be porous. If the whole particle is porous, Grain-Shrinking-Core-Model (GSCM) model provides a good approximation. The disadvantage of SCM and GSCM models is that in each time step along the particle trajectory one or more differential equations must be solved. This costs a lot of computing time.

Postprocessor TRAMIC (unsteady approach, in general processes are not in chemical equilibrium) defines conversion-type and -rate for 40 mineral phases, which have a key role in the transformation of coal minerals into slag under furnace conditions. Most of the kinetic processes applied to the mineral components were determined experimentally at Technical University Braunschweig [6] using advanced equipment (REM, EDX, XRD, TGA, DTA, Mößbauer spectroscopy and other). The gas temperature and gas composition in the furnace control the selection of the kinetic models. Differences between forward and back reactions are considered. The difference of partial pressure between the gas component in the solid particle and the same gas component in surrounding atmosphere of furnace influences the direction of the kinetic reaction f. e. oxidation or reduction. If the description of the mineral matter transformation is given in differential form, the Fehlberg-Method of Runge-Kutta fifth step was successfully applied.

Table 2: Simple mineral transformations [7]

Symbol	Process	f(X)	g(X)
R1	1D PBC	1.0	X
R2	2D PBC	$2(1-X)^{1/2}$	$1-(1-X)^{1/2}$
R3	3D PBC	$3(1-X)^{2/3}$	$1-(1-X)^{1/3}$
D1	1D Diffusion	$\frac{1}{2X}$	X^2
D2	2D Diffusion	$-\frac{1}{\ln(1-X)}$	X + (1-X) ln(1-X)
D3	3D Jander - Type	$\frac{3(1-X)^{2/3}}{2\left(1-(1-X)^{1/3}\right)}$	$(1-(1-X)^{1/3})^2$
D4	3D Ginstling – Brounshtein – Type	$\frac{3}{2}\cdot\frac{1}{(1-X)^{-1/3}-1}$	$1-(2/3)X-(1-X)^{2/3}$
D5	3D Carte - Type $z = V_P/V_A$	$\frac{[1+(z-1)X]^{1/3}(1-X)^{1/3}}{[1+(z-1)X]^{1/3}-(1-X)}$	$[(1+(z-1)X)^{2/3}+(z-1)(1-X)^{2/3}-z]$
An	Nucleation and nuclei's growth JMA, JMAK ($0.5 < n < 4$)	$n(1-X)[-\ln(1-X)]^{\frac{n-1}{n}}$	$[-\ln(1-X)]^{1/n}$
F1	Formal kinetics n^{th} order	-	$(1-X)^n$
-	melting fraction mass/volume	$X_m = \frac{1}{m_{p,0}}\int_0^t\left(\frac{\dot{Q}_{con}+\dot{Q}_{rad}+\dot{S}_{chem}}{\Delta h_m}\right)dt$	

Table 3: Mineral transformation by coupling of physical/chemical processes [8]

Shrinking core model (SCM)

$X = P_n(K\,t)$ $\qquad g(X) = [1 + 2(1-X) - 3(1-X)^{2/3}]$

Generalised shrinking - core model (SCM) for combined chemical reaction, gas-solid (or gas-liquid) mass transfer and multidiffusion. (solid –1 gas-2)

$$X = 1-(r/R_c)^3 \qquad r = \int_0^{R_c}\left(\frac{dr}{dt}\right)dt \qquad \frac{dr}{dt} = -\frac{\nu_{1,E}}{\nu_{2,E}}\frac{M_{2,E(g)}}{\rho_1}k_{eff}C_{2(g)} \qquad k_{eff} = \frac{1}{\frac{1}{k_{chem}}+\left(\frac{r}{R_c}\right)^2\frac{1}{k_b}+\frac{r}{D_p^e}\left(1-\frac{r}{R_c}\right)}$$

GSCM (Grain shrinking - core model)

$$\left(\frac{d^2c}{dR^2}\right)+\left[\frac{2}{R}+\frac{1}{D_p^e}\left(\frac{dD_p^e}{dR}\right)\right]\left(\frac{dc}{dR}\right)-N/D_p^e=0 \qquad R=R_p \qquad D_p^e\left(\frac{dc}{d\bar{R}}\right)=k_b(c_b-c_g) \qquad \bar{R}=\frac{R}{R_p}$$

$$N = k_{eff}c^n A; \qquad R=0 \qquad dc/d\bar{R}=0 \;;$$

volume fraction of solid component 1 $\qquad X_1 = 1-3\int_0^1\left(\frac{M_1\nu_1}{M_1\nu_1+M_2\nu_2}\right)^3\bar{R}^2d\bar{R}$

3. Simulation of test-case

The combustion of bitumenous Saar coal in a test furnace of the IFRF IJmuiden equipped with a jet burner type A1 was simulated to test the mathematical model for forecasting the slagging tendency. The lower heating value of the coal was LHV = 31 MJ/kg and the thermal capacity of the test furnace is 1,825 MW. The pulverised coal, the primary and secondary air enter the test furnace through the burner in the front wall. The flue gas leaves the furnace through an opening in the rear wall. The geometry of the furnace was simplified to be a cuboid of 6,25 m length, 1,87 m height and 2 width and was discretised into 39 x 24 x 38 = 35568 control volumes. With assistance of RRSB-particle size-distribution-function determined by tests all coal particles were divided into ten diameter sizes. The coal mass flow of class correspond to 10 % of the total mass flow at the burner entry. Each diameter class is represented by an average diameter. Using Lagrange method 16000 trajectories including all diameter classes were calculated. Each trajectory is loaded with a particle flow so that the sum of the trajectories mass flow complies with the total coal mass flow.

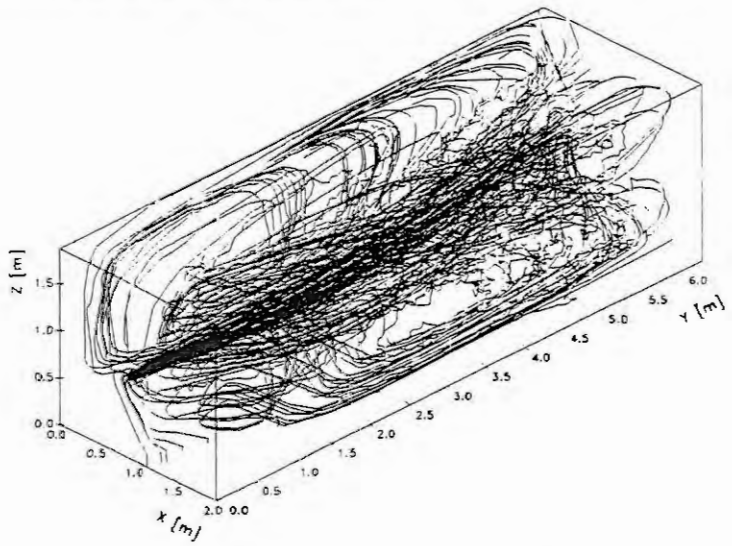


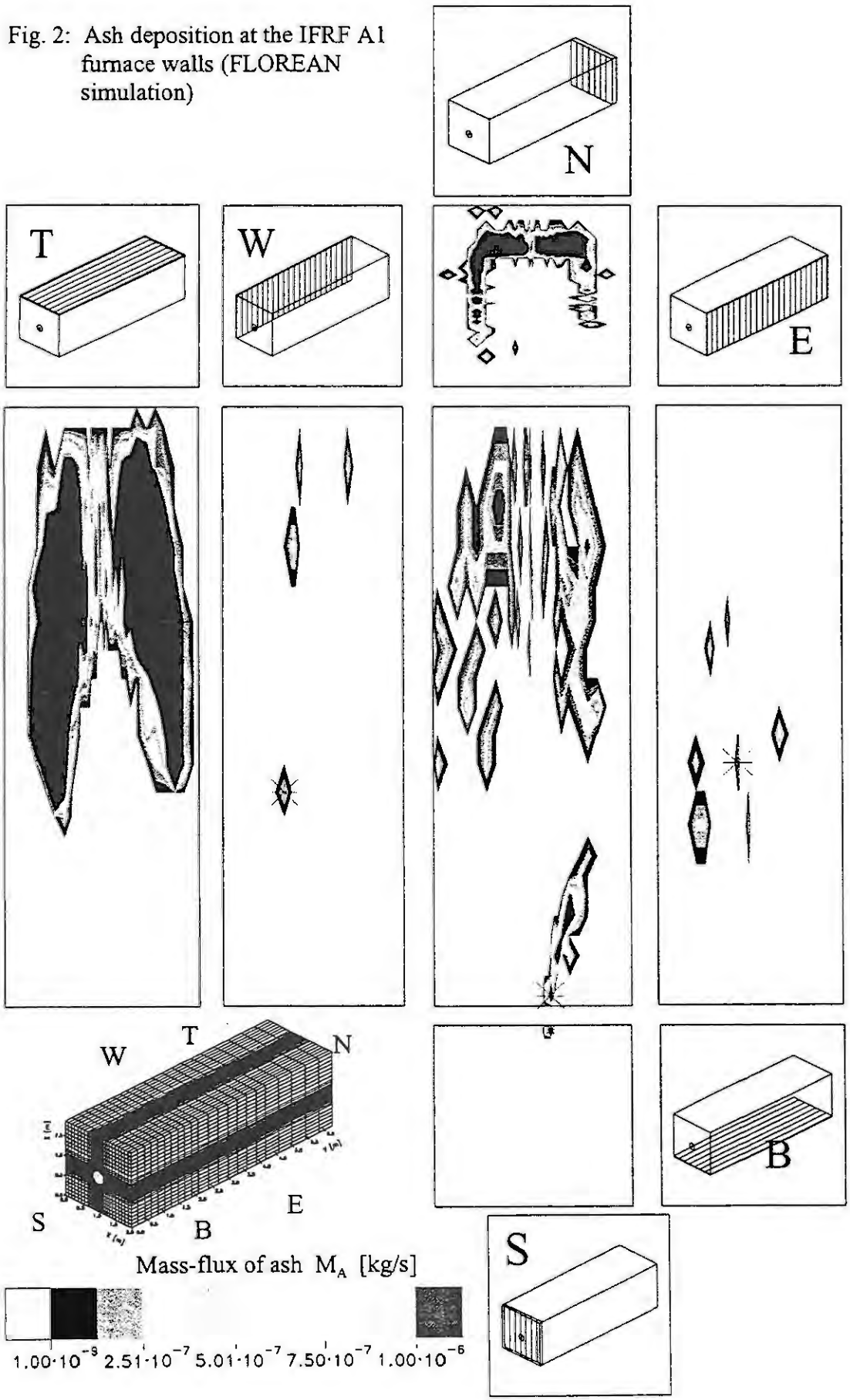
Fig. 1: Trajectories of 200 sticking particles – IFRF A1 furnace.

The combustible coal particle in this simulation contains only one mineral nucleus, which mass is proportional to the ash content of the coal in the proximate coal analysis. In addition it was assumed that the coal structure is porous and permits access of combustion gases to the mineral nucleus without resistance. This assumption is valid only at the trajectory section which includes the coal combustion. When the coal is burnt the remaining mineral nucleus is in direct contact with the furnace atmosphere. Another assumption is that the mineral particle is composed from one or two mineral matter. For the time being 21 different combinations at the start position into burner mouth are possible.

The applied postprocessor TRAMIC includes a mineral distribution model which determines for every coal particle corresponding mineral composition. The number of mineral particles corresponds to the mass ratio of the mineral matter in the coal. At each time step on the trajectory the energy balance is calculated and particle temperature is determined. If the particle temperature at the moment of the particle impact at the furnace wall is higher then softening temperature, the particle sticks and becomes part of the deposit material. In the simulated example out of 16000 calculated trajectories 2752 particles stick at the walls (17,2 %). Dependent from partial pressure of oxygen, water vapor, sulphur and heating rate of particle (also cooling rate), different reactions of the mineral matter of the particle occur on the trajectories.

A further postprocessor WANDMI-AUS determines the mass, coordinates and mineral composition of sticking particle in the moment of impact at the wall. Inside each control volume at the wall, masses of all sticking particles are summarised, total and for each mineral component. Total mass-flux of the fresh slag (kg/s) at all furnace walls is shown at the fig. 2 for simulated test case. It can be also represented as mass-flux-density (kg/m^2s). Fig. 3 shows mass distribution of some important mineral contents at the rear wall of furnace with outlet to chimney. All figures are plotted with the graphical postprocessor D3CLICK. Reasons caused by the geometry of the furnace and the burner type lead to the formation of a strongly recirculating swirl along the axis of the furnace in the upper part. The gas swirl carries the clouds of light particles with it. Zones of particle sticking correspond to zones of higher heat loads (kW/m^3), which generate sticking particles through higher temperature. Visible deposition will be built up at the rear wall above the outlet and at the top wall near to the rear wall.

Fig. 2: Ash deposition at the IFRF A1 furnace walls (FLOREAN simulation)



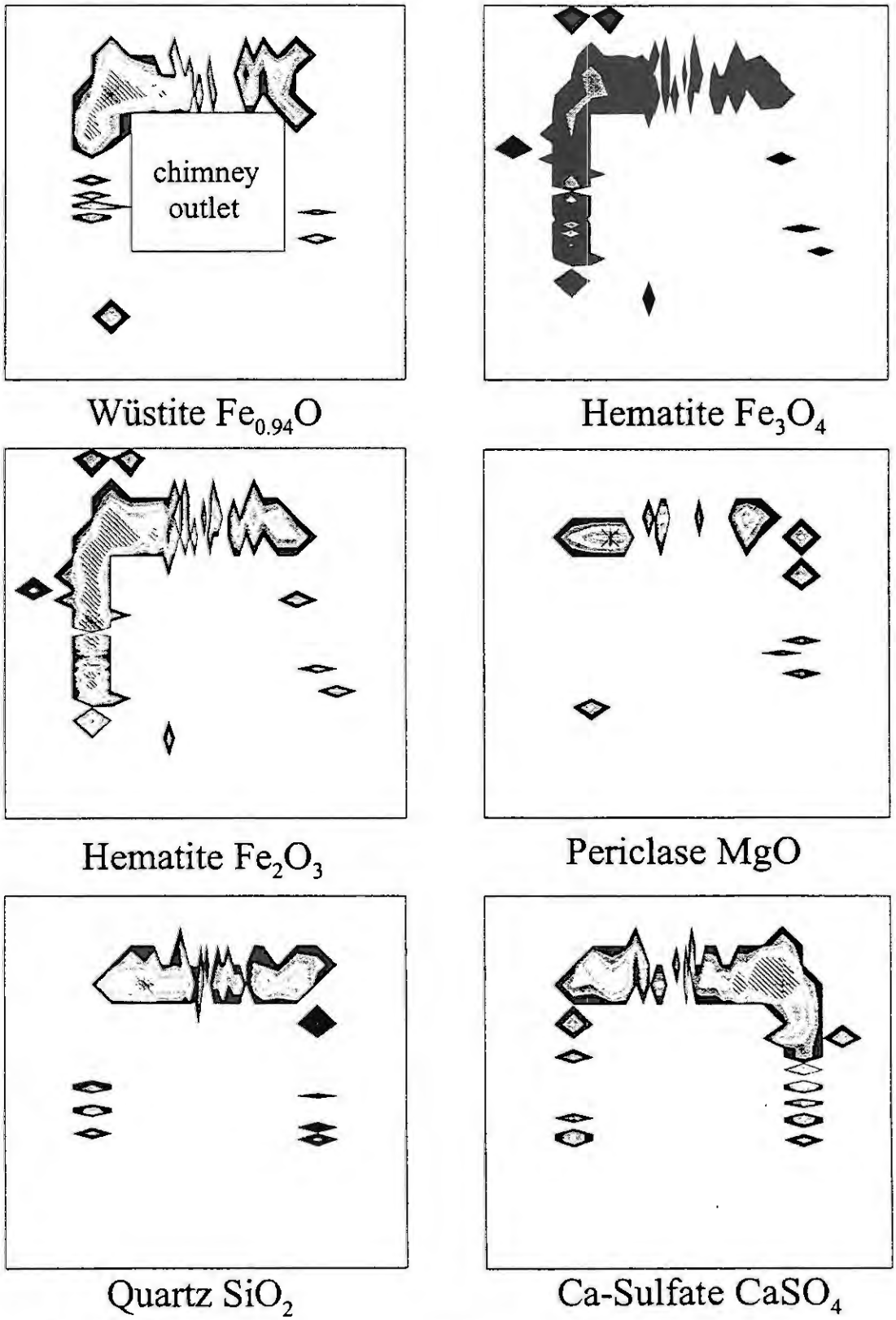


Fig. 3: Distribution of mineral phases at the north wall

4. Summary

The procedure which was developed for forecasting of slagging tendency using furnace simulation program FLOREAN combined with postprocessor TRAMIC, allows a true impression about the deposition quantity and mineral distribution on the furnaces walls. This result takes in to account the coal type, the mineral matter composition in the coal, burner type and furnace geometry, heat loads and corresponding temperature- and concentrations fields of the flue gas along representative coal particle trajectories. A quantitative comparison of the simulation results with measurements was possible only for the gas phase and they correspond well. Even if IFRF furnace is one of most tested furnaces in Europe, comparison for mineral distribution at the wall was not possible, because no measurements exist. For comparison between simulation and reality only a few photos about the wall deposition were available. They permit a coarse qualitative comparison and this shows an agreement.

Developed modelling procedure can be further improved through introduction of:

- advanced coal-mineral models, which takes in to account the fact that a coal particle can consist of numerous mineral matters,
- better sticking criterion, when the particle hits the wall,
- additional reaction systems and models possible in the mineral nucleus of coal particles,
- mineral transformation inside the wall deposits.

5. Acknowledgements

This work is part of the research project AIF 11548 realised with support of the Bundesministerium für Wirtschaft (BMWi) – of the Federal Republic of Germany and the Arbeitsgemeinschaft industrieller Forschungsvereinigungen „Otto von Guericke" e.V., Köln (AiF) and Deutsche Vereinigung für Verbrennungsforschung e.V., Essen. The final research report will be available at the TU Braunschweig – IWBT after 1.10.2000.

6. References

1. H. Müller
Numerische Berechnung dreidimensionaler turbulenter Strömungen in Dampferzeugern mit Wärmeübergang und chemischen Reaktionen am Beispiel des SNCR-Verfahrens und der Kohleverbrennung, VDI-Fortschrittberichte, Reihe 6, Nr. 268, VDI-Verlag GmbH, Düsseldorf, 1992

2. K. C. Fischer, R. Leithner, H. Müller
Three-dimensional simulation of the gas-solid flow in coal-dust fired furnaces,
International Symposium on Two-Phase Flow Modelling and Experimentation,
The Assembly of World Conferences on Experimental Heat Transfer, Fluid Mechanics and Thermodynamics, Rome, Italy, 09th -11th October 1995, Edizioni ETS, Pisa, 1995, p 1387-1393,

3. H. Müller, R.J. Heitmüller
Untersuchung der Brennkammerverschmutzung mit einem mathematischen Modell
VGB-Fachtagung „Feuerungen 1997", Essen, 8./9. 10. 1997; VGB- Kraftwerkstechnik, Heft 2/98

4. O. Bozic, H. Müller, A. Schiller, W. Päuker, R. Leithner
Simulation von Kohlestaubfeuerungen einschließlich Verschlackung unter besonderer Berücksichtigung der Pyritumwandlung, VGB-Konferenz „Forschung für die Kraftwerkstechnik 1998",
11th –12th Februar 1998, Essen, Tagungsband TB 233A

5. O. Bozic, H. Müller, R. Leithner
Simulation der Verschlackung von braunkohlegefeuerten Brennkammern
VDI-Geselschaft Energietechnik, Fachtagung „Forschrittliche Braunkohlenutzung im liberalisierten Strommarkt", 23./24. February 1999, Cottbus, VDI Berichte No. 1456

6. K. D. Becker, S. Kipp, A. Renwrantz, F. Schrobsdorff
Experimentelle Untersuchungen zum Projekt AIF 10639 – Phase 1-3 (Final Report),
TU Braunschweig - IPTC, 1998

7. H. Tanaka, Thermal analysis and kinetics of solid state reactions, Thermochimica Acta 267, 1995, p. 29 - 44

8. L. D. Smooth (Editor), Fundamentals of coal combustion – for clean and efficient use, Elsevier, 1993

9. J. B. Michel, R. Payne
Detailed measurement of long pulverized coal flames for the characterization of pollutant formation, International Flame Research Foundation, IJmuiden, December 1980

Nomenclature

Latin Letters

Symbol	Unit	Definition
A	m^2	surface,
b_{ij}	-	coefficient,
B_i	kW/m^2	radiative flux component,
c	kg/m^3	concentration,
C_i	$kmol/m^3$	concentration,
C	-	empirical constant,
D	m^2/s	diffusion coefficient,
E_i	$kJ/kmol$	activation energy,
f_i	-	factor,
h	kJ/kg	specific enthalpy,
k	m^2/s^2	turbulent kinetic energy,
k_i		reaction coefficient, depending on reaction
K	1/s	coefficient of transformation ,
K_a	m^{-1}	absorption coefficient,
m	kg	mass,
M	$kg/kmol$	molecular mass of component i,
n	-	order of reaction,
Q	kJ	heat,
$\dot{Q}$	kW	heat flux,
Pn		polinom
$\Re$	$kJ/kmol{\cdot}K$	general gas constant
r	m	internal particle radius (SCM model),
R	m	internal particle radius (GSCM model),
R_c	m	outer particle radius (SCM model),
R_p	m	outer particle radius (GSCM model),
S_ϕ		sink- / source term, depending on ϕ
t	s	time,
T	K	temperature,
u	m/s	gas velocity,
u_i	m/s	gas velocity component,
V	m^3	volume,
Y_i	kg/kg	mass fraction of species i,
x_i	m	coordinate direction,
X	m^3/m^3	volume fraction,

Greek Symbols

Symbol	Unit	Definition
Δh_m	kJ/kg	melting specific enthalpy,
Γ_ϕ		molecular exchange coefficient, depending of ϕ (f. e. 1, k, ε, h..)
δ_{ij}	-	Kronecker – δ_{ij}
ε	m^2/s^3	turbulent dissipation rate,
ε_{por}	-	porosity,
μ	$kg/(ms)$	dynamic viscosity,
$\nu_{i,r}$	-	stoichiometric coefficient of species i in reaction r,
π	-	constant,
ρ	kg/m^3	density,
σ	kW/m^2K^4	Stefan – Boltzman constant,
σ_i	-	Schmidt – Prandtl number,
ϕ		common variable gas, (f. e. k, ε, h, Y_n)

Subscripts

b	boundary layer
E	educts
chem	chemical
con	convection
eff	effective
g	gas
i,j,k	indices
p	particle
por	porosity
P	products
r	number of reaction
rad	radiation
spe	species
t	turbulent
0	start value

Superscripts

e	effective value
~	actual value
-	time-averaged value
´	turbulent fluctuation value
n	max. number of educts
m	max. number of products

FINITE ELEMENT MODELLING OF MOORING LINES

O.M. Aamo and T.I. Fossen
Department of Engineering Cybernetics
Norwegian University of Science and Technology
N-7491 Trondheim, Norway

Abstract In this paper, we develop a new finite element model for a cable suspended in water. Global existence and uniqueness of solutions of the truncated system is shown for a slightly simplified equation describing the motion of a cable having negligible added mass and supported by fixed end-points. Based on this, along with well known results on local existence and uniqueness of solutions for symmetrizable hyperbolic systems, we conjecture a global result for the initial-boundary value problem.

1 Introduction

This paper is a subset of another paper [1] dealing with position mooring systems (PM) for offshore oil production. PM systems have been commercially available since the late 1980's, and have proven to be a cost-effective alternative to permanent platforms for offshore oil production. In traditional testing of the performance of PM systems by means of computer simulations, tabulated static solutions of the cable equation have been coupled to the vessel dynamics. This approach is adequate for shallow waters. However, in deeper waters, dynamic interactions between the vessel and mooring system renders such a quasi-static approach inaccurate [3].

Software packages that solve the cable equation by means of the finite element method (FEM) are readily available. However, such general purpose FEM packages are not suited for control system design, and are usually slower than software tailored for a particular application. Moreover, the theoretical aspects, such as existence and uniqueness of solutions, are often taken for granted. In fact, FEM tools were developed and used, for instance in structural engineering, decades before a sound theoretical foundation was established [4].

In this paper, a new finite element model of a cable suspended in water is derived. The hydrodynamic loads on the cable are modelled according to *Morison's equation* (see for instance, [2]). For a slightly simplified equation, describing the motion of a cable having negligible added mass and supported by two fixed end-points, we show global existence and uniqueness of solutions of the truncated system, and conjecture a global result for the initial-boundary value problem.

2 PDE for the cable dynamics

The equation of motion of a cable with negligible bending and torsional stiffness is given by (see for instance [6])

$$\rho_0 \frac{\partial \vec{\mathrm{v}}(t,s)}{\partial t} = \frac{\partial}{\partial s}(T(t,s)\vec{\mathrm{t}}(t,s)) + \vec{\mathrm{f}}(t,s)(1+e(t,s))$$

where t is the time variable, and $s \in [0,L]$, $\vec{\mathrm{v}}:[t_0,\infty)\times[0,L]\to\mathbb{R}^3$ and $\vec{\mathrm{t}}:[t_0,\infty)\times[0,L]\to\mathbb{R}^3$ are distance along the unstretched cable, velocity and tangential vector, respectively. L is the length of the unstretched cable, ρ_0 is mass per unit length of unstretched cable, $T:[t_0,\infty)\times[0,L]\to\mathbb{R}$ is tension, $e:[t_0,\infty)\times[0,L]\to\mathbb{R}$ is strain and $\vec{\mathrm{f}}:[t_0,\infty)\times[0,L]\to\mathbb{R}^3$ is the sum of external forces (per unit length of unstreched cable) acting on the cable. By introducing the position vector $\vec{\mathrm{r}}:[t_0,\infty)\times[0,L]\to\mathbb{R}^3$, we get $\vec{\mathrm{t}}=\frac{1}{1+e}\frac{\partial\vec{\mathrm{r}}}{\partial s}$ such that

$$\rho_0 \frac{\partial^2 \vec{\mathrm{r}}}{\partial t^2} = \frac{\partial}{\partial s}\left(\frac{T}{1+e}\frac{\partial \vec{\mathrm{r}}}{\partial s}\right) + \vec{\mathrm{f}}(1+e)$$

Applying *Hooke's law* yields

$$\rho_0 \frac{\partial^2 \vec{r}}{\partial t^2} = \frac{\partial}{\partial s}\left(EA_0 \frac{e}{1+e}\frac{\partial \vec{r}}{\partial s}\right) + \vec{f}(1+e)$$

where E is *Young's modulus* and A_0 is the cross-sectional area of the unstretched cable.

2.1 External forces

In addition to gravity, a submerged cable is subject to hydrostatic and hydrodynamic forces, i.e.

$$\vec{f} = \vec{f}_{(hg)} + \vec{f}_{(dt)} + \vec{f}_{(dn)} + \vec{f}_{(mn)}$$

where $\vec{f}_{(hg)}$ constitutes the bouyancy (gravity and hydrostatic) force per unit length of unstretched cable, $\vec{f}_{(dt)}$ and $\vec{f}_{(dn)}$ are tangential and normal hydrodynamic drag, respectively, per unit length of unstretched cable and $\vec{f}_{(mn)}$ is the hydrodynamic inertia force per unit length of unstretched cable.

Gravity and hydrostatic forces

It is assumed that we can regard each element of the cable as completely surrounded by water so that

$$\vec{f}_{(hg)} = \rho_0 \frac{\rho_c - \rho_w}{(1+e)\rho_c}\vec{g}$$

where $\vec{g} \in \mathbb{R}^3$ is the gravitational acceleration, ρ_c is density of the cable and ρ_w is density of the ambient water.

Hydrodynamic forces

From *Morison's equation,* see for instance [2], we get the following expression for hydrodynamic drag per unit length of unstretched cable

$$\begin{aligned}
\vec{f}_{(dt)} &= -\frac{1}{2}C_{DT}d\rho_w \left|\vec{v}\cdot\vec{t}\right|(\vec{v}\cdot\vec{t})\vec{t} = -\frac{1}{2}C_{DT}d\rho_w |\vec{v}_t|\,\vec{v}_t \\
\vec{f}_{(dn)} &= -\frac{1}{2}C_{DN}d\rho_w \left|\vec{v}-(\vec{v}\cdot\vec{t})\vec{t}\right|(\vec{v}-(\vec{v}\cdot\vec{t})\vec{t}) = -\frac{1}{2}C_{DN}d\rho_w |\vec{v}_n|\,\vec{v}_n
\end{aligned}$$

where C_{DT} and C_{DN} are tangential and normal drag coefficients for the cable, respectively, and d is the cable diameter. The hydrodynamic inertia force per unit length of unstretched cable is given by:

$$\vec{f}_{(mn)} = -C_{MN}\frac{\pi d^2}{4}\rho_w(\vec{a}-(\vec{a}\cdot\vec{t})\vec{t}) = -C_{MN}\frac{\pi d^2}{4}\rho_w\vec{a}_n$$

where C_{MN} is a hydrodynamic mass coefficient and $\vec{a}:[t_0,\infty)\times[0,L] \to \mathbb{R}^3$ is the acceleration. The subscripts n and t on $\vec{v}$ and $\vec{a}$ denote decompositions into the normal and tangential directions, respectively.

Formulation of the initial-boundary value problem

We have the following initial-boundary value problem

$$\rho_0 \frac{\partial^2 \vec{r}}{\partial t^2} - \frac{\partial}{\partial s}\left(EA_0 \frac{e}{1+e}\frac{\partial \vec{r}}{\partial s}\right) - (1+e)\left(\vec{f}_{(hg)} + \vec{f}_{(dt)} + \vec{f}_{(dn)} + \vec{f}_{(mn)}\right) = 0 \tag{1}$$

with boundary conditions

$$\vec{r}(t,0) = \vec{r}_0(0),\ \ \vec{r}(t,L) = \vec{r}_0(L),\ \ \text{for all } t \geq t_0$$

and initial conditions

$$\vec{r}(t_0,s) = \vec{r}_0(s),\ \ \vec{v}(t_0,s) = \vec{v}_0(s)$$

Here, $\vec{r}_0 : [0,L] \to \mathbb{R}^3$ and $\vec{v}_0 : [0,L] \to \mathbb{R}^3$ are initial cable configuration and initial cable velocity, respectively.

3 Discretization into finite elements

Discretization of the initial-boundary value problem is performed using the Galerkin method and finite elements. This method consists of the following steps

1. The initial-boundary value problem (1) is transformed into the corresponding *generalized problem*. This is done by multiplying the equation by the functions $\vec{w} \in \mathcal{V}$, and then integrating by parts over $[0, L]$. $\mathcal{V}$ is a suitable space of functions in which to search for a solution.
2. Restriction of $\vec{r}$ and $\vec{w}$ to appropriate finite-dimensional subspaces $\mathbb{V}_n \subset \mathcal{V}$, yields the Galerkin method.
3. Choosing the finite-dimensional subspaces such that they are spanned by bases consisting of so-called finite elements, yields a particularly simple set of ordinary differential equations. This is the finite element method.

The Galerkin equation resulting from (1) is given by

$$\frac{\rho_0 l}{6}(\ddot{\mathbf{r}}_{k-1} + 4\ddot{\mathbf{r}}_k + \ddot{\mathbf{r}}_{k+1}) + EA_0\left[\frac{e_k}{\varepsilon_k}\mathbf{l}_k - \frac{e_{k+1}}{\varepsilon_{k+1}}\mathbf{l}_{k+1}\right] = \int_0^L \left(\vec{\mathbf{f}}_{(hg)} + \vec{\mathbf{f}}_{(dt)} + \vec{\mathbf{f}}_{(dn)} + \vec{\mathbf{f}}_{(mn)}\right)(1+e)\varphi_k ds \qquad (2)$$

$$k = 1, 2, ..., n-1$$

where

$$\begin{aligned}
\mathbf{l}_k &= \mathbf{r}_k - \mathbf{r}_{k-1} \\
e_k &= \frac{1}{l}|\mathbf{r}_k - \mathbf{r}_{k-1}| - 1 = \frac{|\mathbf{l}_k|}{l} - 1 \\
\varepsilon_k &= l(1+e_k) = |\mathbf{l}_k| \\
\varphi_i(s) &= \begin{cases} 0 & s < (i-1)l \\ \frac{1}{l}s - (i-1) & (i-1)l \le s < il \\ -\frac{1}{l}s + i + 1 & il \le s < (i+1)l \\ 0 & (i+1)l \le s \end{cases}, \quad i = 0, 1, 2, ..., n
\end{aligned}$$

The subscripts (hg), (dt), (dn) and (mn) stand for hydrostatic and gravity forces, tangential drag forces, normal drag forces, and hydrodynamic added inertia forces, respectively. n is the number of finite elements, and $l = L/n$ is the unstretched length of each element. The triangular form of the φ_i functions reflects the choice of a finite element basis for the subspaces $\mathbb{V}_n$. Note that in this form, algebraic expressions for the drag forces cannot be found. However, in Section 5, approximations are introduced that eliminate the need for numerical integration of these terms.

4 Existence and uniqueness of solutions

In this section we show existence and uniqueness of solutions for a slightly simplified equation under the assumption of strictly positive strain. This is the main contribution of the paper.

Assumption 1 *There exists a constant $c > 0$, such that*

i) $e(t, s) \ge c$ *for* $s \in [0, L]$ *and for all* $t \ge t_0$, *and;*
ii) $e_k(t) \ge c$ *for* $k = 1, 2, ..., n$ *and for all* $t \ge t_0$.

Neglecting the added mass term $\vec{\mathbf{f}}_{(mn)}$, which means that we assume drag dominant behaviour, and considering a damping term in the form

$$\vec{\mathbf{f}}_{(d)} = -\frac{1}{2}C_D d\rho_w |\vec{\mathbf{v}}|\,\vec{\mathbf{v}}$$

yield the following slightly modified initial-boundary value problem

$$\frac{\partial^2 \vec{r}}{\partial t^2} - \frac{EA_0}{\rho_0}\frac{\partial}{\partial s}\left(\frac{e}{1+e}\frac{\partial \vec{r}}{\partial s}\right) - \frac{\rho_c - \rho_w}{\rho_c}\vec{g} + \frac{1}{2\rho_0}(1+e)C_D d\rho_w |\vec{v}|\,\vec{v} = 0 \tag{3}$$

with boundary conditions

$$\vec{r}(t,0) = \vec{r}_0(0), \quad \vec{r}(t,L) = \vec{r}_0(L), \quad \forall\, t \geq t_0$$

and initial conditions

$$\vec{r}(t_0,s) = \vec{r}_0(s), \quad \vec{v}(t_0,s) = \vec{v}_0(s)$$

Our goal is to apply Proposition 2.1 in [5, page 370], which states local existence and uniqueness of solutions for symmetrizable hyperbolic systems. Thus, we need to show that equation (3) is symmetrizable. Define $\mathbf{u}(t,s)$ as follows

$$\mathbf{u}(t,s) = \begin{bmatrix} \mathbf{u}_0 \\ \mathbf{u}_1 \\ \mathbf{u}_2 \end{bmatrix} \triangleq \begin{bmatrix} \vec{r} \\ \frac{\partial \vec{r}}{\partial s} \\ \frac{\partial \vec{r}}{\partial t} \end{bmatrix}$$

Carrying out the differentiation in the first term on the right hand side of (3), the equation, in terms of $\mathbf{u}$, can be written as

$$\mathbf{A}_0\frac{\partial \mathbf{u}}{\partial t} = \mathbf{A}_1\frac{\partial \mathbf{u}}{\partial s} + \mathbf{g} \tag{4}$$

where

$$\begin{aligned}
\mathbf{A}_0(t,s,\mathbf{u}) &= \begin{bmatrix} \mathbf{I} & 0 & 0 \\ 0 & \frac{EA_0}{\rho_0}\left(\frac{\mathbf{u}_1\mathbf{u}_1^T}{(1+e)^3} + \frac{e}{1+e}\mathbf{I}\right) & 0 \\ 0 & 0 & \mathbf{I} \end{bmatrix} \\
\mathbf{A}_1(t,s,\mathbf{u}) &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \frac{EA_0}{\rho_0}\left(\frac{\mathbf{u}_1\mathbf{u}_1^T}{(1+e)^3} + \frac{e}{1+e}\mathbf{I}\right) \\ 0 & \frac{EA_0}{\rho_0}\left(\frac{\mathbf{u}_1\mathbf{u}_1^T}{(1+e)^3} + \frac{e}{1+e}\mathbf{I}\right) & 0 \end{bmatrix} \\
\mathbf{g}(t,s,\mathbf{u}) &= \begin{bmatrix} \mathbf{u}_2 \\ 0 \\ \frac{\rho_c - \rho_w}{\rho_c}\vec{g} - \frac{1}{2\rho_0}(1+e)C_D d\rho_w |\mathbf{u}_2|\,\mathbf{u}_2 \end{bmatrix}
\end{aligned}$$

In (4), we have multiplied the equation by the matrix $\mathbf{A}_0$, which is symmetric positive definite for $s \in [0,L]$ and for all $t \geq t_0$ under Assumption 1 (in fact, there exists a constant c, such that $\mathbf{A}_0 \geq c\mathbf{I} > 0$, $s \in [0,L]$, $\forall t \geq t_0$). Notice that the matrix $\mathbf{A}_1$ is rendered symmetric, by means of the *symmetrizer* $\mathbf{A}_0$. Thus, Proposition 2.1 in [5, page 370], provides local existence of a unique solution to (4). However, based on the following arguments, we will conjecture that the solution can be continued for all time. For the Galerkin equation corresponding to (3), which is given by

$$\begin{aligned}
\frac{\rho_0 l}{6}(\ddot{\mathbf{r}}_{k-1} + 4\ddot{\mathbf{r}}_k + \ddot{\mathbf{r}}_{k+1}) + EA_0\left[\frac{e_k}{\varepsilon_k}\mathbf{l}_k - \frac{e_{k+1}}{\varepsilon_{k+1}}\mathbf{l}_{k+1}\right] &= \int_0^L \left(\vec{\mathbf{f}}_{(hg)} + \vec{\mathbf{f}}_{(d)}\right)(1+e)\varphi_k ds \qquad (5) \\
k &= 1,2,\ldots,n-1
\end{aligned}$$

we can state the following result (proven in [1]).

Theorem 1 *For any $n \in \{2,3,4,\ldots\}$, let the initial state $(\mathbf{r}(t_0), \mathbf{v}(t_0)) = (\mathbf{r}_0, \mathbf{v}_0)$ be given. If Assumption 1 holds, then there exists a unique solution of (5) for all $t \geq t_0$.*

Theorem 1 implies that $\mathbf{u}_n \in H^k([0,L])$, for $t \geq t_0$, where $H^k([0,L])$ denotes the Sobolev space defined by

$$H^k([0,L]) = \left\{ \mathrm{u} \in L^2([0,L]) \;\middle|\; \frac{\partial^l \mathbf{u}}{\partial s^l} \in L^2([0,L]),\ 0 < l \leq k \right\}$$

with the natural norm

$$\|\mathbf{u}\|_{H^k([0,L])} = \sum_{l=0}^{l=k} \left\| \frac{\partial^l \mathbf{u}}{\partial s^l} \right\|_{L^2([0,L])}$$

and $\mathbf{u}_n$ is given in terms of the finite element basis defined in Section 3, that is

$$\mathbf{u}_n(t,s) = \sum_{i=0}^{n} \begin{bmatrix} \mathrm{r}_i(t)\varphi_i(s) \\ \mathrm{r}_i(t)\frac{\partial \varphi_i(s)}{\partial s} \\ \mathrm{v}_i(t)\varphi_i(s) \end{bmatrix} \tag{6}$$

In fact, Theorem 1 implies that there exists a constant c, independent of k and n, such that

$$\|\mathbf{u}_n\|_{H^k([0,L])} \leq c, \quad \forall t \geq t_0, \quad n = 2, 3, \ldots$$

Based on the above considerations, along with the results of Chapter 16, Sections 1 and 2 in [5, pages 359-372], we conjecture the following.

Conjecture 1 *Suppose Assumption 1 holds, and that* $\mathbf{u}(0,s) \in H^k([0,L])$, *with* $k \geq 2$. *Then there exists a unique solution* $\mathbf{u} \in C([t_0,\infty), H^k([0,L]))$, *to the initial-boundary value problem (4). Moreover, the sequence of solutions* $\mathbf{u}_n$ *(as given in (6)) of the Galerkin equation (5), converges to* $\mathbf{u}$ *in the following sense*

$$\|\mathbf{u} - \mathbf{u}_n\|_{H^k([0,L])} \to 0 \text{ as } n \to \infty$$

Remark 2 *We stress the fact that since Theorem 1 and Conjecture 1 are stated under Assumption 1, global solutions are not guaranteed for all initial conditions. The problem of finding conditions on the initial data under which Assumption 1 holds (for all* $t \geq t_0$*), is outside the scope of this work.*

5 Implementation

It is desirable to apply certain approximations to the terms of equation (2) in order to simplify implementation. Looking at the k^{th} node, we see by inspection of equation (2), that it takes an advantageous form if the following approximations are applied

$$\begin{aligned} \dot{\mathbf{r}}_{k-1} &\approx \dot{\mathbf{r}}_k, \quad \dot{\mathbf{r}}_{k+1} \approx \dot{\mathbf{r}}_k \\ \ddot{\mathbf{r}}_{k-1} &\approx \ddot{\mathbf{r}}_k, \quad \ddot{\mathbf{r}}_{k+1} \approx \ddot{\mathbf{r}}_k \end{aligned}$$

With these approximations, equation (2) reduces to the following:

$$\left[\left(\rho_0 l + \frac{C_1}{2}(\varepsilon_k + \varepsilon_{k+1}) \right) \mathrm{I}_{3\times 3} - \frac{C_1}{2} \left(\frac{\mathrm{l}_k \mathrm{l}_k^T}{\varepsilon_k} + \frac{\mathrm{l}_{k+1} \mathrm{l}_{k+1}^T}{\varepsilon_{k+1}} \right) \right] \ddot{\mathbf{r}}_k = \mathrm{f}_{k_{(hg)}} + \mathrm{f}_{k_{(dt)}} + \mathrm{f}_{k_{(dn)}} + \mathrm{f}_{k_{(\tau)}}, \quad k = 1, 2, \ldots, n-1 \tag{7}$$

where

$$
\begin{aligned}
\mathbf{f}_{k_{(r)}} &= EA_0 \left[\frac{e_{k+1}}{\varepsilon_{k+1}} \mathbf{l}_{k+1} - \frac{e_k}{\varepsilon_k} \mathbf{l}_k \right] \\
\mathbf{f}_{k_{(hg)}} &= l\rho_0 \frac{\rho_c - \rho_w}{\rho_c} \begin{bmatrix} 0 & 0 & g \end{bmatrix}^T \\
\mathbf{f}_{k_{(dt)}} &= -\frac{C_2}{2} \left[|\dot{\mathbf{r}}_k \cdot \mathbf{l}_k| \frac{\mathbf{l}_k \mathbf{l}_k^T}{\varepsilon_k^2} + |\dot{\mathbf{r}}_k \cdot \mathbf{l}_{k+1}| \frac{\mathbf{l}_{k+1} \mathbf{l}_{k+1}^T}{\varepsilon_{k+1}^2} \right] \dot{\mathbf{r}}_k \\
\mathbf{f}_{k_{(dn)}} &= -\frac{C_3}{2} \left[\varepsilon_k \left| \left(\mathbf{I}_{3\times 3} - \frac{\mathbf{l}_k \mathbf{l}_k^T}{\varepsilon_k^2} \right) \dot{\mathbf{r}}_k \right| \left(\mathbf{I}_{3\times 3} - \frac{\mathbf{l}_k \mathbf{l}_k^T}{\varepsilon_k^2} \right) \right. \\
&\quad \left. + \varepsilon_{k+1} \left| \left(\mathbf{I}_{3\times 3} - \frac{\mathbf{l}_{k+1} \mathbf{l}_{k+1}^T}{\varepsilon_{k+1}^2} \right) \dot{\mathbf{r}}_k \right| \left(\mathbf{I}_{3\times 3} - \frac{\mathbf{l}_{k+1} \mathbf{l}_{k+1}^T}{\varepsilon_{k+1}^2} \right) \right] \dot{\mathbf{r}}_k \\
C_1 &= C_{MN} \frac{\pi d^2}{4} \rho_w, \quad C_2 = \frac{1}{2} C_{DT} d \rho_w, \quad C_3 = \frac{1}{2} C_{DN} d \rho_w
\end{aligned}
$$

$\mathbf{I}_{3\times 3}$ is the 3×3 identity matrix, and the subscript (r) stands for internal reaction forces. Clearly, in the limit as $n \to \infty$, (2) and (7) are identical. Modelling a moored vessel is now a matter of assembling the above equations for each mooring line. The details of this procedure are available in [1].

6 Conclusions

In this paper, we have developed a new finite element model for a cable suspended in water. Global existence and uniqueness of solutions of the truncated system is shown for a slightly simplified equation describing the motion of a cable with negligible added mass and supported by fixed end-points. Based on this, along with well known results on local existence and uniqueness of solutions for symmetrizable hyperbolic systems, we conjecture a global result for the initial-boundary value problem.

7 Acknowledgements

This work was supported by the Research Council of Norway, which is gratefully acknowledged. The first author would also like to thank Professor Helge Holden, Department of Mathematical Sciences, Norwegian University of Science and Technology, for his helpful comments to Section 4, and Dr. Jann Peter Strand, ABB Industri AS, for his general comments, and support of the project.

References

[1] Aamo, O. M. and Fossen, T. I. Finite element modelling of moored vessels. *Submitted to the Journal of Mathematical and Computer Modelling of Dynamical Systems*, 2000.

[2] Faltinsen, O. M. *Sea Loads on Ships and Offshore Structures.* Cambridge University Press, 1990.

[3] Ormberg, H., Fylling, I. J., Larsen, K., and Sødal, N. Coupled analysis of vessel motions and mooring and riser system dynamics. In *Proc. of the 16th Int. Conf. on Offshore Mechanics and Arctic Engineering*, pages 91–100, New York, 1997. American Society of Mechanical Engineers.

[4] Strang, G. and Fix, G. J. *An analysis of the finite element method.* Prentice-Hall, Inc., 1973.

[5] Taylor, M. E. *Partial Differential Equations III.* Springer-Verlag New York, Inc., 1996.

[6] Triantafyllou, M. S. Cable mechanics with marine applications, lecture notes. Department of Ocean Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139, USA, May 1990.

ADAPTING BLOCK METHOD TO SOLVE MOIST AIR FLOW MODEL

Monika Woloszyn*(1) **Gilles Rusaouën*** **Jean-Jacques Roux*** **Thierry Dagusé****
* Cethil-ETB, INSA de Lyon, bât. 307, 20, av. A. Einstein, 69621 Villeurbanne Cedex, France
** EDF-DER, Dept. ADE Bâtiments, BP 1, 77250 Moret-sur-Loing, France
(1) tel. 33.4.72.43.84.62, fax. 33.4.72.43.85.22, e-mail :woloszyn@insa-cethil-etb.insa-lyon.fr

Abstract. Coupled problems describing global thermal behaviour of buildings are difficult to solve. We are interested here in the numerical resolution of algebraic non-linear systems corresponding to the steady state behaviour. After a short description of the physical problem, main ideas of block methods are presented. Then splitting of the analysed model is discussed and the performance of block methods is evaluated. Finally, a robust hybrid block strategy, well adapted to moist airflow modelling, is elaborated.

Introduction

Solving systems of non-linear equations is a necessary step in simulating real actions. However, this task is often difficult. Particularly, coupled systems of equations are usually hard to solve, because of complex physical interactions. In such situations, block strategies can be helpful to find the desired numerical solution. In practice, the performance of different block algorithms varies, according to the physical problem.

We are interested here in solving a system of algebraic non-linear equations describing steady state of moisture-energy-airflow model. The model is implemented in *CLIM2000* simulation environment, developed by Electricité de France [1]. *CLIM2000* solves simultaneously the whole system of equations and this approach is obviously not adapted to moist air models. The numerical methods used often diverge. Actually, in practice, even simpler energy-airflow models are found difficult to solve, and often block methods are needed to compute the solution [4] [7].

First part of this paper introduces the main characteristics of the system of equations. In the second part, block methods are presented and splitting of the analysed model is discussed. Then the block methods are applied to solve our model using two different update strategies for unknowns.

System of equations

Multizone models, where the air volume of one room is represented by one node are of interest here. In practice, systems of equations describing global energy-moisture-airflow behaviour of multizone space are difficult to solve. Most difficulties are due to the simplified airflow model, very sensitive to the variations of energy and moisture values. In order to understand the relationship between the model and the numerical method, this research is limited to basic configurations representing moist air movements inside a partitioned space. The results presented here refer to the configuration in figure 1. The equations are mainly three types of balances :

- dry air mass balance : determining air movements (variable: air pressure at ground level),
- vapour mass balance : describing vapour transfers in a multizone space (variable: air moisture content),
- enthalpy balance : energy calculations enabling temperature computations (variable: air temperature).

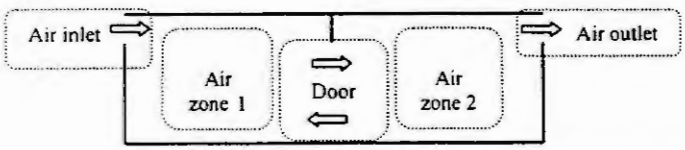


Figure 1. Analysed configuration

A detailed example of the system of equations can be found in [8]. Here, only the most important points are evoked. Algebraic non-linearities of resulting equations are mainly due to two factors :

- air flow equations, linking pressure difference (ΔP, [Pa]) and the mass flow (Q, [kg/s]) using the power law: $Q = K \Delta P^n$, (K and n : power law real coefficients),
- perfect gas relationship: $PV = MrT$; linking pressure (P, [Pa]), temperature (T, [K]) and mass (M, [kg]) with air volume (V, [m^3]) and perfect gas constant (r, [J/(kg K)]).

We are concerned here with steady state description, therefore the whole problem can be written as a system of non-linear algebraic equations: $F(x)=0$.

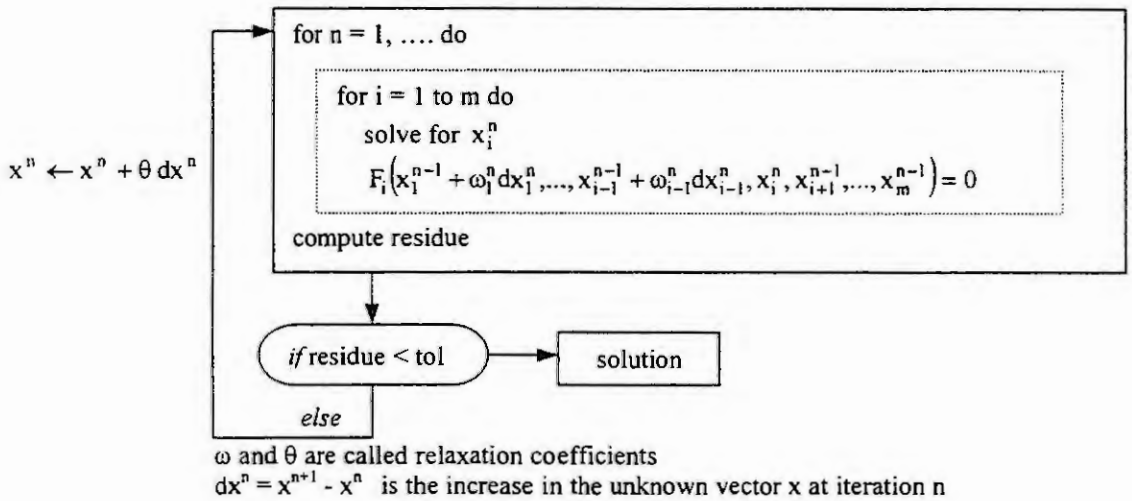
The important sensitivity of the coupled model to the variations of boundary conditions is mainly due to the representation of the airflow through large openings (such as doors). We use here the popular simplified expression based on Bernoulli equation [3]. In such configurations different flow directions are possible : 1→2, 2→1 and even a two-way flow : 1↔2. Integration of different possibilities and of their transitions into the airflow model introduces situations difficult to solve. Indeed, the airflow model is very sensitive to even small variations of boundary conditions. Typically, an important difference of the mass flow of about 100 kg/h can be introduced by a very small difference of pressures of about 10^{-2} 10^{-3} Pa. At the moment, no robust method to compute the solution of the complete problem exists. Indeed, popular non-linear solvers are based on Newton's direction [2] [5], where at each non-linear iteration we need to solve the linear system:

$$J(x)\, dx = -F(x) \qquad (1)$$

The jacobian matrix, J(x), is ill-conditioned. Even for very small systems, such as the configuration presented in figure 1, the associated matrix has a condition number of about 10^9.

Block methods

Dividing a large, difficult problem, into smaller and easier to treat, is often the only method to overcome resolution difficulties. This type of methods, called 'block methods', can be applied to solve systems of non-linear algebraic equations. The function F(x) is the whole system of equations composed of m blocks. A general block-based strategy to solve F(x) = 0 is presented in figure 2.



ω and θ are called relaxation coefficients

$dx^n = x^{n+1} - x^n$ is the increase in the unknown vector x at iteration n

Figure 2. General block method for non-linear systems

Following analogy with linear systems, block method with ω=1 and θ=0 is called Gauss-Seidel method, and with ω=0 and θ=1 Jacobi method. Relaxation coefficients ω and θ can be constant, or may vary with the block number and iteration. Therefore, many different combinations of block methods are possible.

Splitting physical problem into blocks

In order to apply a block strategy on our problem, the whole system of equations must be split into subsystems. In the construction of a general method, adapted to a large family of building simulations problems, an efficient splitting must be based on physical criteria. In our case two main possibilities can be chosen : (1) splitting using location criteria (per 'room' : one block is formed by equations describing the behaviour of one room) or (2) splitting using phenomenological criteria (per 'system' one block is composed of equations describing one physical system : energy, moisture or airflow).

The final choice should satisfy two types of constrains :

- numerical: easy resolution of each block,
- physical: easy application to the whole family of models.

Some interesting conclusions can be driven from the study of jacobian matrix' structure. The condition numbers of different blocks are computed using the ℓ_2 norm, on the example presented in figure 1, for different values of physical parameters (external climate, position of outlets and inlets ...).

block	complete matrix	airflow block	energy block	moisture block	room block
condition number	10^7 - 10^9	$1 - 10^3$	$10^1 - 10^2$	$10^1 - 10^2$	$10^6 - 10^8$

Table 1. Condition number of different blocks of the jacobian matrix.

In the table 1, the superiority of splitting into 'systems' can be easily seen. The condition number corresponding to one physical system is low, because one block contains one type of physical balances. Condition number of room block is almost as high as the condition number of the whole matrix, because they both include different types of equations (dry air, vapour and energy balances).

In addition, the results in table 1 suggest that the numerical resolution of each physical subsystem should be easy. First numerical tests show the practical superiority of system decomposition. Moreover this approach can be easily generalised to building simulation tools. In the next, only the results concerning decomposition into systems are presented.

The order of blocks is defined using physical relations between variables. Two slightly different orders are used here : moisture-energy-airflow and energy-moisture-airflow both with a pre-resolution of the airflow block.

To solve each bloc we use the popular Newton-LU method with a line search procedure to help global convergence [5]. In all cases we start with the initial vector $\bar{0}$. The demanded tolerance on the residue (euclidian norm) is of 10^{-9} for each block and 10^{-6} for the global residue.

Application of Gauss-Seidel update

Gauss-Seidel update is one of the most popular among block strategies. It is applied on our system, keeping its partition into blocks. For example, all the moisture balances are solved simultaneously, and then moisture contents values are updated in energy and airflow blocks. This strategy is applied to the configuration presented in figure 1. Main results are regrouped in table 2. Each case correspond to different parameter values (external climate, inlet flow, height of air outlet).

case	global iterations	final residual	observations
1	62	$1.9\ 10^{-3}$	demanded precision not reached
2	71	$1.5\ 10^{-3}$	
3	100	$8\ 10^{-3}$	
4	19	$9.6\ 10^{-5}$	
5	3	$1.2\ 10^{-4}$	

Table 2. Solving using Gauss-Seidel update between blocks

Each block is easily solved, however global iterations are not converging, even when an important number is done. In all cases the iterations approach the solution, however they stagnate and the demanded precision is impossible to reach.

In order to design a method fitted to our problem, the origin of this situation must be understood. Actually, in our model, the enthalpy flow $\dot{H}$ can be written as :

$$\dot{H} = h(\theta)\, \dot{m}_{dry\ air}(\theta) \qquad (2)$$

where h is mass enthalpy of moist air [J/kg], $\dot{m}_{dry\ air}$ is mass flow [kg/s] and θ is air temperature [°C]

Both h and $\dot{m}_{dry\ air}$ depend upon the air temperature θ, which is actually the mathematical variable of our problem. Solving F(x)=0 in the energy block means in fact finding the temperature value to balance the enthalpy flows. This can be done changing either h or $\dot{m}_{dry\ air}$ values. In the treated case, using full Gauss-Seidel iteration, $\dot{m}_{dry\ air}$ varies in a more important way than h. This prevents the system from global convergence.

Adapting updates to physical interactions

This last observation must be taken into consideration in the construction of a method efficient on our problem. Numerical iterations should fix dry air mass flow during the resolution of energy block. The enthalpy flow should be computed using the following equation (3) instead of (2) :

$$\dot{H} = h(\theta^n)\, \dot{m}_{dry\ air}(\theta^{n-1}) \qquad (3)$$

This is a hybrid method. Gauss-Seidel update is used only for the transported enthalpy, and the flow term value computed during the previous iteration is maintained.

The adapted updates are then applied on our problem. We use two slightly different variants of block method (energy and moisture blocks are permuted). Typical results are presented in tables 3 and 4.

global iteration	moisture block		energy block		airflow block	
	iterations	global resid.	iterations	global resid.	iterations	global resid.
0	-	-	-	-	3	0.1
1	1	0.5	2	0.4	3	$3\ 10^{-9}$

Table 3. Application of block method. Iterations needed to solve one block and global residual

global iteration	energy block		moisture block		airflow block	
	iterations	global resid.	iterations	global resid.	iterations	global resid.
0	-	-	-	-	3	0.1
1	2	0.01	1	0.2	2	0.08
2	2	0.2	0	0.2	2	$1\ 10^{-9}$

Table 4. Application of block method. Iterations needed to solve one block and global residual

For all cases treated (different values of physical parameters) the behaviour of the block method is very similar. Adaptation of the updates to the physical interactions results in a rapid global convergence. The best strategy converge in one iteration and the number of iterations needed to compute each block is stable.

Conclusions and perspectives

Block methods for non-linear system are very efficient. However they must be adapted to the physical problem solved. In our case good understanding of the physical problem leads to a very efficient block algorithm, combining adequate block decomposition with hybrid method to update values of unknowns. The performance of this hybrid method is mainly due to good transcription of physical interactions into numerical iterations.

The proposed method is well adapted to treated problems. Moreover, comprehensive adapting of numerical iterations to physical system can be easily generalised. It can be directly applied to some other building physics' delicate problems, such as pollutants propagation and could be also extended to other fields. After its validation, this method is intended to be implemented in the *CLIM2000*'s solver.

References

1. Bonneau, D., Rongere, F.X., Covalet, D., Gauthier, B. Clim2000 : Modular Software for Energy Simulation in Buildings. In *IBPSA 93*. Adelaïde (Australia), 1993. p 19-25.
2. Dennis, J.E., Schnabel R.B. *Numerical methods for unconstrained optimization and nonlinear equations*. Philadelphia (USA) : SIAM, 1996. 2nd ed. 378 p.
3. IEA Annex XX. *Air flow patterns within buildings*. Report Annex XX, Subtask 2 : Air flows between zones. Air flow through large openings in buildings. Ed. J. van der Maas. Lausanne (Suisse) : International Energy Agency. Energy Conservation in Buildings and Community Systems Programme, 1992, 163 p.
4. Hensen, J. Modelling coupled heat and air flow : ping pong vs onions. *Proc. 16th AIVC Conf.*, Palm Springs, Sept. 1995, 8 p.
5. NETLIB [On-line] Knoxville (USA) : UTK. [02.07.99]. Avaliable from internet <URL : http://www.netlib.org>
6. Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P. *Numerical recipes in Fortran. The art of scientific computing*. 2nd ed., Cambridge (USA) : Cambridge University Press, 1992. 933 p.
7. Schneider, P.S. *Air flow - thermal behaviour of buildings : strategies to solve the coupled problem*. (in French) PhD thesis : INSA de Lyon (France), 1994.
8. Woloszyn, M. *Moisture-energy-airflow modelling of multizone buildings: a strategy proposed to solve the integrated system of equations*. (in French) PhD thesis : INSA de Lyon (France), 1999.

THE FLOW PROBLEM IN HEATED TUBE-HEADER-STRUCTURES

H. Walter, K. Ponweiser and W. Linzer
Institute of Thermal Engineering,
Vienna University of Technology,
Getreidemarkt 9, A-1060 Vienna

Abstract. For the design of a water tube boiler, which is planned to operate under dynamic conditions, the knowledge of the behaviour of the transient fluid flow in the network structure of the steam generator is of great importance. For the prediction of this dynamic behaviour numerical methods are suitable.

In this paper a tube-collector-model is presented, which is appropriate for the creation of the mathematical equations describing the fluid flow in a network of tubes. The difference between an explicite and an implicite calculation method is shown. Special attention is directed to the boundary conditions at both ends of the tubes.

Finally some results of the simulation of the start-up of a Heat Recovery Steam Generator (HRSG) are presented.

Introduction

Since the late 70's not only computer programs for thermal design are used, but also software for the calculation of the fluid flow distribution at steady state in ramified tube-systems have been developed [1].

The reliability of operation of natural circulation boilers as well as boilers with forced circulation and once-through boilers can be improved, if the distribution of the fluid flow in the complex pipe system under different load conditions is known already during design state.

Due to an augmented use of steam generators for peak load generation as well as for waste heat utilisation, the requirements on the dynamic behaviour of these plants increase. In literature many articles about the dynamic simulation of steam generators can be found. The scope of these studies covers models for single components as well as computer codes, which are able to calculate the whole dynamic behaviour of a power plant (a detailed review is given by [2]). In case of different load change velocities at parallel arranged heat surfaces it is important to be able to forecast the dynamic change of the mass flow distribution in the ramified pipe system.

For this intention programs based on a tube-collector-model are useful, such as that one presented in this paper.

Formulation of the problem

High power steam generators as well as smaller units used in industrial power plants, for example Heat Recovery Steam Generators (HRSG) arranged behind a gas turbine, are built as water tube boilers. In these boilers the water and water/steam mixture has to pass through tubes with different heat absorption, which are connected with distributors and collectors.

Advantages of boilers with natural circulation evaporators are the reduced investment, maintenance and operation costs due to the absence of a circulation pump. Figure 1 shows a HRSG with a natural circulation evaporator. In normal case, the water flows from the downcomer through the heated tube bank straight to the riser relief tubes. The driving force of the natural circulation is due to the difference of the density of the water in the downcomer and the water/steam mixture in the tube bank and the riser relief tubes. The most critical operation modes of a HRSG with natural circulation evaporators are fast cold start-ups and heavy load changes. In these cases stagnation or even reverse flow can occur due to dynamic effects. In order to avoid such situations it is important to have detailed information about the flow distribution in the tube network already in the stage of boiler design. To make available such important data, the Institute of Thermal Engineering at the Vienna University of Technology has developed a computer code for the simulation of the dynamic behaviour of heated tube networks as well as natural circulation boilers.

Modelling

During model creation the physical phenomena under consideration as well as the CPU-performance of the computer used for simulation have to be taken into account. If the whole cycle has to be simulated an overall model, which approximates the certain components of the steam generator in a general way, will be sufficient. If we focus on the flow conditions in different groups of pipes a much more detailed model is necessary.

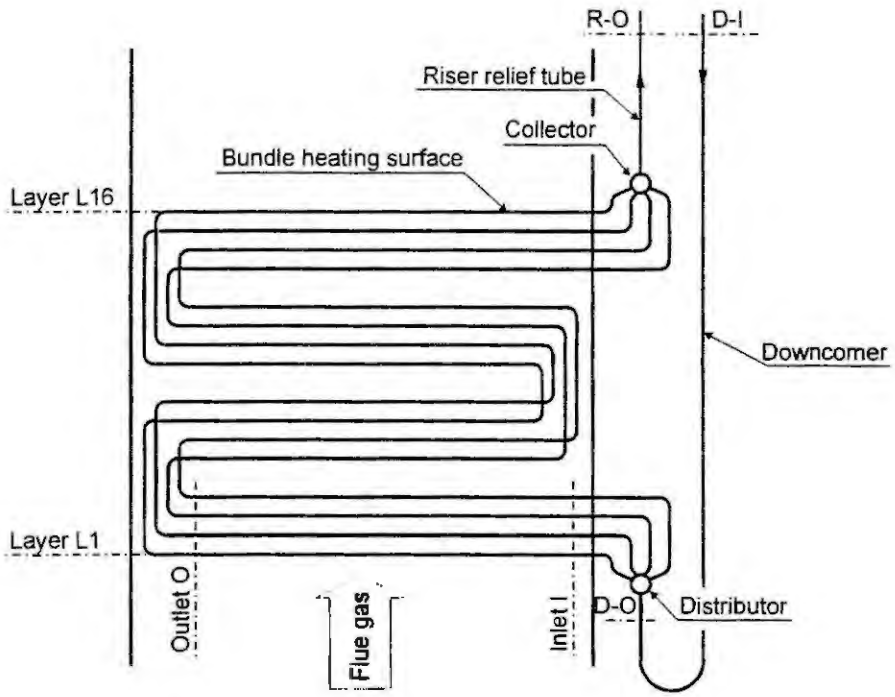


Figure 1: Model of the evaporator of a vertical HRSG

Figure 1 shows the model of a natural circulation evaporator of a vertical HRSG. To analyse the mass flow distribution in different layers of the evaporator the steam generator was subdivided into the following sections: one downcomer, one distributor, four parallel heat surfaces with different heat input for each layer, one collector and one riser relief tube. The sketch of the steam generator in figure 1 does not give the correct image about the horizontal dimensions of the heating surface, which can reach up to 20 m in reality, whereas the vertical division of the bundle is shown correctly.

Model of the tube flow

The mass flow in the tubes of a steam generator can be assumed to be one-dimensional, because they are very long compared to their diameter. In the model under consideration tubes with the same geometry and heat impact are combined to one single tube. This simplification requires, that for the calculation of the mass flow and the heat flux, the inner and outer surfaces as well as the cross sections of the single tubes are multiplied by the number of parallel tubes. But for the calculation of the pressure drop the inner diameter of the single tube must be used.

For a straight tube with constant cross section the unsteady mass balance may be written as:

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho w}{\partial x} = 0 \tag{1}$$

where the coordinates of location and time are x and t respectively. The density ρ and the velocity w are mean values over the cross section of the tube.

The following equation represents the momentum balance, which postulates that the temporary change of momentum is equal to the local change of the momentum flux and the forces, which are acting on the fluid volume:

$$\frac{\partial \rho w}{\partial t} + \frac{\partial \rho w w}{\partial x} = -\frac{\partial p}{\partial x} - \rho g_x + \left(\frac{\partial p}{\partial x}\right)_{friction} \tag{2}$$

In this equation, p means the pressure and g_x is the component of the gravitation in the direction of the tube axis.

Considering the fluid flow in steam boilers, the thermal energy is much higher than the kinetic energy as well as the potential energy and the expansion work. Therefore, the balance of the thermal energy can be simplified to:

$$\frac{\partial \rho h}{\partial t} + \frac{\partial \rho w h}{\partial x} = \dot{q}\frac{U}{A} \tag{3}$$

In this equation, h denotes the specific enthalpy and $\dot{q}$ the heat flux referred to the surface of the liquid in the tube. To fulfil the balance, $\dot{q}$ must be multiplied by the ratio perimeter U over cross section A of the liquid in the tube.

Discretization and solution of the governing equations

Basically, both explicit and implicit methods can be used to solve the differential equations describing the fluid flow in a tube. Explicit calculation methods are characterised by the following relationships: the state in a discrete volume at location x_i and at a discrete time t_n is a function of the state of the same volume at time t_{n-1} as well as of the state of the neighbour volumes at the locations x_{i-1} and x_{i+1}, both at time t_{n-1} ($t_{n-1} = t_n - \Delta t$, $x_{i-1} = x_i - \Delta x$ and $x_{i+1} = x_i + \Delta x$).

In implicit calculation methods the state in one discrete volume $x_{i,n}$ is influenced by the states in the volumes $x_{0,n} \ldots x_{i-1,n}$ and $x_{i+1,n} \ldots x_{N,n}$ (time level t_n) as well as by the states in the volumes $x_{0,n-1} \ldots x_{N,n-1}$ (time level t_{n-1}).

If the transient behaviour of the fluid flow in a tube has to be simulated with an explicit method, comp. [3], it is possible to calculate the state in the volumes x_i, $i = 2, 3, \ldots, N-1$, if adequate boundary conditions are available to describe the state in the volumes x_1 and x_N at time t_n. Nevertheless, to ensure the stability of the explicit method, the Courant-Friedrich-Levi-condition (CFL-condition) [4], which is limiting the size of the time step depending on the geometric data, has to be fulfilled. Using an explicit method for the simulation of the fluid flow in a tube, the ratio between the step size of geometry and the step size of time has to be smaller than the sum of the velocities of the fluid flow and the speed of sound of the fluid. Because the speed of sound in water is very high the time step has to be chosen very small. As a consequence of that the simulation by using an explicit method is very time consuming.

Using an implicit algorithm only the desired accuracy of the calculation limits the value of the time steps. The disadvantage of this calculation method is, that the system of differential equations must be approximated iteratively for every single time step.

An implicit algorithm to solve the partial differential equations of mass, momentum and energy with a finite-volume-method is presented by Patankar [5], who derives the SIMPLE algorithm (Semi Implicit Method for Pressure Linked Equations) in a general form. An implementation of the SIMPLE algorithm for one-dimensional fluid flow in a tube is described in detail in [2] or [6].

As a result of the discretization, we obtain from the partial differential equations for the momentum an algebraic system of equations for the calculation of the velocity distribution. The partial differential equation of the mass balance linked with the differential equation of the momentum balance yield in an algebraic system of equations for the calculation of a pressure correction from one iteration step to the next. The differential equation for the energy results in an algebraic system of equations for the calculation of the enthalpy. The state in a discrete tube section depends directly on the state in the neighbour volumes of the tube. So the three algebraic systems of equations have a tridiagonal matrix of coefficients:

$$\begin{bmatrix} a_{1,P} & -a_{1,E} & & & \\ -a_{2,W} & a_{2,P} & -a_{2,E} & & \\ & \cdots & \cdots & \cdots & \\ & & -a_{N-1,W} & a_{N-1,P} & -a_{N-1,E} \\ & & & -a_{N,W} & a_{N,P} \end{bmatrix} \cdot \begin{bmatrix} \varphi_1 \\ \varphi_2 \\ \cdots \\ \varphi_{N-1} \\ \varphi_N \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \cdots \\ b_{N-1} \\ b_N \end{bmatrix} \tag{4}$$

For certain sections of discretization the solution vector φ includes the velocity w in the momentum balance, the values for the pressure correction p^* in the system of the pressure correction equations and the enthalpy h in the energy balance. Methods for the calculation of the coefficients a and the vector of the constants b are given in [2], [5] or [6].

Boundary conditions

At the tube inlet two characteristic lines ($\lambda_1 = w + c$, $\lambda_2 = w$ and $\lambda_3 = w - c$, whereby w denotes the fluid velocity and c the speed of sound in the fluid) run into the tube if the fluid velocity is positive at this point. Due to that fact, two boundary conditions must be set at this tube end. At the tube outlet one characteristic line leaves the tube if the fluid velocity is positive at this point. Therefore one boundary condition is necessary.

For explicit methods, as the method of characteristics, algebraic equations are needed for the calculation of the state quantities inside of the tube (x_2 to x_{N-1}) at the time level t_n. At the same time level, additional equations are needed to determine that state quantities at the boundaries (x_1 and x_N) which are not given as boundary values.

If the SIMPLE algorithm is used for the calculation, it is possible to include the desired boundary conditions in the coefficient matrices. If e. g. the enthalpy is given at the tube inlet the coefficients of the energy balance have to be set to $a_{1,P} = 1$ and $a_{1,E} = 0$. This selection causes a remaining coefficient h_1, while the influence on its neighbour volume (element 2) will be preserved. The same method is used at the tube outlet. The pressure as boundary condition can be taken into account in the coefficient matrix of the pressure correction equations. A given velocity of the fluid at one tube end can be considered in the system of equations for the momentum. Considering the physical requirements, two boundary conditions have to be set on the tube end where the fluid enters the tube, while for the tube end where the fluid leaves the tube one boundary condition is sufficient.

For forced circulation boilers and once through boilers, as they are explored in [2], it is rather efficient to give at the tube inlet the mass flow and the velocity respectively and the enthalpy as boundary condition while at the outlet the pressure must be given as boundary condition. The pressure at the tube inlet as well as the mass flow and the thermal conditions of the fluid at the tube outlet are results of the calculation.

For natural circulation boilers and other pressure driven systems it is better to determine the pressure at both ends of the tube and to predefine additionally the state of the fluid at that tube end where the fluid enters the tube. In this case the mass flows or the velocities respectively at both ends of the tube and the state of the fluid at the tube end where mass leaves the tube are results of the calculation.

Using this boundary conditions, it is very important, that the numerical treatment of the fluid flow problem is absolute symmetrical, otherwise a change of the flow direction due to the pressure boundary conditions cannot be handled.

Model of the collector

Assuming that the distribution of the thermodynamic state in the collector is homogeneous, for the calculation the collector can be seen as one single point. This assumption is admissible, because the vertical dimension of the collector is small compared to that of the remaining tube system. So the gravity distribution of density and pressure can be neglected. The huge differences of the cross sections between the collector and its connected tubes are responsible for strong turbulence, so that a segregation of the fluid in the collector will not occur.

Because the collectors are assumed to be nods the equations for the mass and energy balance are ordinary *differential equations with time t as independent variable:*

$$\frac{d}{dt}\rho_S V_S = \sum_j \rho_j w_j A_j - \sum_k \rho_k w_k A_k \tag{5}$$

$$\frac{d}{dt}\rho_S h_S V_S = \sum_j \rho_j w_j h_j A_j - \sum_k \rho_k w_k h_k A_k \tag{6}$$

The variables of the collector are denoted with the index S; j represents values at the collector inlet and k denotes the values at the outlet. V_S is the volume of the collector and A the cross section of the connected tubes. Analogous to the treatment of the fluid flow in the tube, kinetic energy as well as expansion work are neglected in the energy balance.

Because momentum has vectorial nature, considering flow in a straight tube, it acts in the direction of the tube axis, which can be seen in the momentum balance (Eq. 2). If several tubes are connected to the collector from different directions, the momentum fluxes must not be added arithmetically but rather vectorially.

The velocity of the fluid in the collector is rather small compared to that inside the tubes. So it can be assumed, that the momentum of the fluid will be lost at the entrance of the collector and has to be rebuilt at the outlet of it. Based on this assumption, the momentum balance of the collector is reduced to a pressure balance.

The changes of the momentum at the inlet and the outlet can be taken into account using a pressure loss coefficient ζ:

$$p_S = p_j - \frac{\zeta_j}{2}\rho_j w_j |w_j| \qquad \text{and} \qquad p_S = p_k + \frac{\zeta_k}{2}\rho_k w_k |w_k| \tag{7}$$

Inclusion of the conservation equations of the collector into the equations of the fluid flow

After the discretization of the differential equations, the collectors are reduced to nods in the grid structure of time and space.

Using explicit calculation methods, the density of the fluid in the collector at the time t_n can be computed by means of the mass balance of the fluid in the collector and the mass flows in each single connection point between the tubes and the collector at the time t_{n-1}. The energy balance provides the inner energy of the fluid in the collector at the time t_n, by means of the energy of the fluid in the collector and the energy flow at the tube connecting points of the collector. By use of the internal energy it is possible to calculate the remaining thermodynamic properties, in particular the pressure inside of the collector. With the collector pressure at the time t_n (Eq. 7), the pressure at the connecting points between tubes and collector can be determined, which is one of the required boundary condition for each single tube. At the outlet connection points between collector and tube, where fluid is leaving the collector, an additional boundary condition has to be defined. That boundary condition is the specific enthalpy h, which is calculated from the pressure loss of the fluid leaving the collector, assuming adiabatic throttling with constant total enthalpy:

$$h_S = h_j + 0.5 w_j^2 \quad \text{for } w_j < 0, \text{ and} \qquad h_S = h_k + 0.5 w_k^2 \quad \text{for } w_k > 0 \tag{8}$$

If $w_j > 0$ and $w_k < 0$ h_j and h_k are results of the calculation of the fluid flow in the tube. These values must not to be changed in the balance equations of the collector.

The calculation of the collector pressure using the method described above is possible only if the time steps remain small. Experience shows, that a certain ratio between the collector volume and the volume of the connected tube element should not be exceeded, taking also into consideration the width of time steps given by the CFL-condition.

Summarised the overall calculation procedure for an explicit simulation of the fluid flow in a tube network can be subdivided into the following four points:

1. Calculation of the thermodynamic properties in the collectors for time step n, using the values of the time step n-1.
2. Computation of the boundary conditions for the fluid flow depending on the thermodynamic properties inside of the collectors.
3. Stepwise calculation of the thermodynamic properties at the discrete points of the tubes.
4. Calculation of the thermodynamic properties for the next time step, starting at 1.

Implicit calculation methods have the advantage that the time increments can be of any size. Due to that fact they are be preferred in the case of more complex tube-collector-systems and will be discussed in detail therefore.

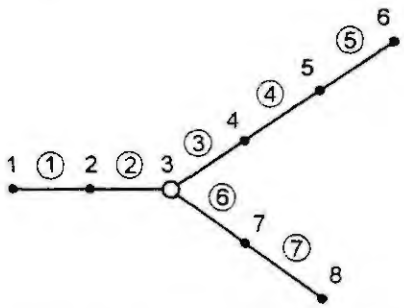


Figure 2: Discretization of a tube-collector-connection

If an implicit calculation method, for example the SIMPLE--algorithm, is used, the calculation of the state variables in a single tube at a certain time step cannot be decoupled from the state variables in the collector and

the other tubes. So the discretized differential equations for the collector consist of an equation for mass, momentum and energy conservation, whereby each of them represents one line in the system of equations.

Figure 2 illustrates for instance three tubes which are connected to one collector. Two of the tubes are subdivided into two cells whereas the third tube is divided into three elements. In the figure the elements of the tubes are labelled by circled numbers, the nodes by numbers only. The values in the nodes of the elements represent the averaged values of the state variables of the control-volumes. The velocities are related to the interfaces of the control-volume (so called staggered grid). The node of the collector is treated in the same way as any node of the tube, but with the difference that it has more adjacent cells than a node in a single cell of a tube.

Assuming that the interruption of the flow of momentum at the connection points between tubes and collector causes a pressure drop, the algebraic system of equations, which describes the velocity distribution in the tube network, is decoupled at the collectors. The system of equations remains tridiagonal, zero elements are found in the upper and lower diagonal at the place of decoupling. The structure of the coefficient matrix is presented in the following left scheme. The system of equations for the calculation of the pressure correction looses its tridiagonal structure in the case of a network with tube-collector-structure as shown in the right scheme.

$$\begin{bmatrix} x & x & & & & & \\ x & x & & & & & \\ & & x & x & & & \\ & & x & x & x & & \\ & & & x & x & & \\ & & & & & x & x \\ & & & & & x & x \end{bmatrix} \qquad \begin{bmatrix} x & x & & & & & \\ x & x & x & & & & \\ & x & x & x & & & x \\ & & x & x & x & & \\ & & & x & x & x & \\ & & & & x & x & \\ & & x & & & x & x \\ & & & & & x & x \end{bmatrix}$$

The coefficients located outside the tridiagonal field and unequal zero are deduced from the mass balance of those collectors, which are connected to more than two tubes. The control-volumes of the single tubes consecutively numbered and so they form tridiagonal sections in the coefficient matrix.

The distribution of the coefficients in the matrix depends on the way in which order the connected tubes are counted. Therefore it is recommended to optimise the bandwith of the coefficient matrix before starting the numerical solving algorithm.

From the description above can be seen that for the calculation of the pressure correction one more equation is necessary than for the determination of the velocities. This is caused due to the fact that the collector represents an additional node in the system. The set of equations for the calculation of the specific enthalpy h has the same structure as the system of equations for the computation of the pressure correction.

For the description of the network with a tube-collector-structure and for data handling the use of the graph theory is recommended.

Simulation example

In this chapter the simulation of a start-up procedure of a HRSG, will be presented (the basic structure is shown in figure 1). The tube-collector-structure was applied for modelling the steam generator, without taking the drum into consideration. The numerical simulation was done using the SIMPLE-algorithm.

The following initial conditions are assumed:

- The Fluid in the pipe network has constant temperature (near saturation state).
- There is no fluid motion.
- The distribution of pressure and density is due to the gravity.

Boundary conditions:

- At the inlet of the downcomer and at the outlet of the riser relief tube the pressure is equal and constant during simulation.
- The water entering the downcomer or the riser relief tube (only if reverse flow occurs) is assumed to be at saturation state.
- The heat fluxes to the heating surfaces of the bundle are given as functions of time.

The time depend development of the heat flow transferred from the flue gas side was calculated by the use of an external boiler calculation program. The simulation is based on the assumption that flue gas enters at the

bottom and leaves the evaporator at the top. The lowest layer is heated most, the heat flux in the following tube rows decreases in a logarithmic way.

The heat flux starts at time $t = 0$, it is increased linearly for 120 seconds, remains constant up to $t = 420$ seconds, is increased for another time span of 480 seconds linearly and remains constant afterwards.

In figure 3 and 4 the development of the mass flow for certain selected cells of the system are presented as a result of the simulation.

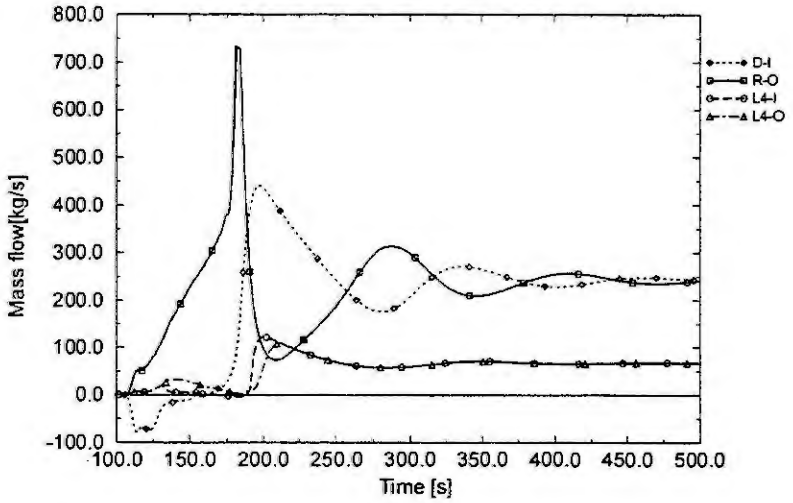


Figure 3: Mass flow in the riser rlief tube, downcomer and layer 4

Figure 3 gives an overview for the first 500 seconds after start up. After about 110 seconds the steam production starts. The start of steam production can be observed by the mass discharge through the outlet of the riser relief tube. The mass discharge is accompanied by reverse flow in the downcomer, discernible on the negative mass flow at the inlet of the downcomer (D-I). Further heating changes the mass flow in the downcomer into positive direction and causes a strong acceleration. During that phase very high mass flow rates can be observed at the outlet of the riser relief tube (R-O), but also the mass flow in the downcomer is remarkable above its stationary value. The difference between the mass flow at the outlet of the riser relief tube and the inlet of the downcomer can be regarded as "ejection of mass". Due to the fact that during that phase so much mass is discharged the remaining amount of mass in the system is smaller than the steady state requires. At about 200 seconds storage of mass starts and after about 500 seconds this process is finished and turns into a steady state.

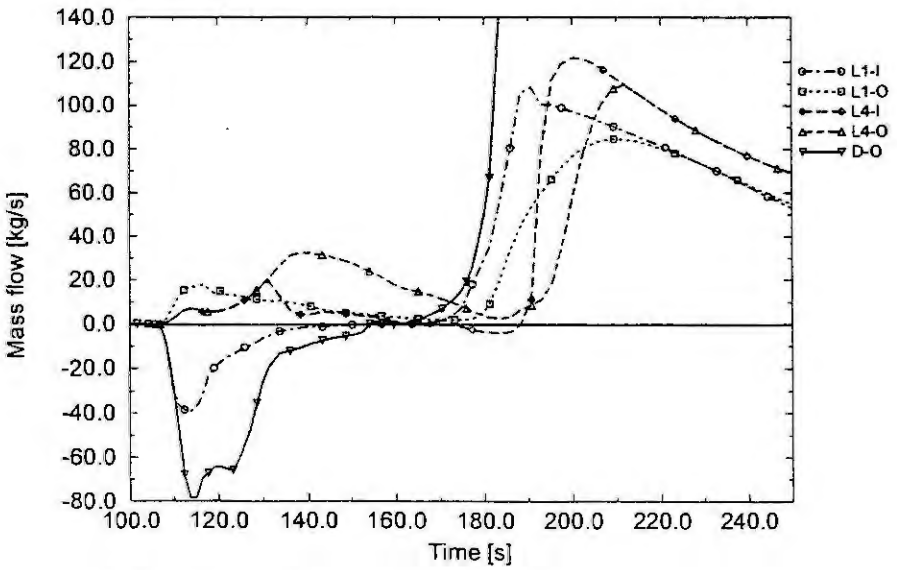


Figure 4: Mass flow in layer 1 and 4 and at the downcomer outlet

The reason for the reverse flow in the downcomer at the first period of steam production can be explained by figure 4. Due to the start of steam generation at approximately stagnant fluid flow a high quantity of water is ejected from the first heated layer (L1). Because of the fact that the tubes of the bundle have a higher pressure loss than the downcomer with the greater diameter, the discharged mass of the first layer flows mainly into the distributor. The negative mass flow at the inlet of the heating surface (L1-I) is almost twice the positive mass flow at the outlet of the heating surface (L1-O), which indicates the above mentioned effect. The heating surfaces 2 and 3 which are not shown in figure 4 have a similar tendency. The fourth heating surface which is heated less and later, shows positive flow direction already before the steam production is starting.

Because of the steam-water-mixture in the evaporator and the riser relief tubes (I-RO) the hydrostatic pressure is smaller at this side of the system than in the downcomer. This pressure difference retards the reverse flow after about 130 seconds and forces the fluid flow into positive direction. Between the 180^{th} and 220^{th} second mass is stored into the showed heating surface layers again. After that period the differences between the inlet and the outlet mass flow from these layers are only very small.

Simulations for different warm starts of the presented HRSG showed that high heat loads during the first phase cause increased reverse flow in the downcomer. As a consequence steam enters the downcomer at the lower end and causes a decrease of fluid flow density. Finally the density of the fluid in the downcomer is lower than in the evaporator-riser-relief-tube part and the change to the desired positive flow direction is not possible anymore.

With the simulation program presented in this paper it is possible to detect even periodical oscillations of mass flow at certain load conditions, which also can be observed in existing plants. After detailed studies these phenomenon could be associated with density wave oscillations, which are the result of the superposition of friction inside the tube and the storing and discharging process of mass.

Conclusion

For the simulation of the fluid flow in a tube network a mathematical model has been developed, which describes the fluid flow in the tubes as well as the changes of state in the connecting elements. The tube-collector-model presented in this paper is very well suited for the simulation of this problem. The fluid flow in the tube is modelled one-dimensionally by means of the conservation equations for the mass, the momentum and the energy. The collectors are modelled as points, whereby the differential equation for the momentum balance is reduced to a pressure balance.

Using an explicit calculation method, the conservation equations for the collector supply the boundary conditions for the calculation of the fluid flow in the tubes. However, if an implicit calculation method (for example the SIMPLE-algorithm) is used, the conservation equations for the collector must be provided in a similar form as the equations for the description of the fluid flow in the tube, in order to enable the integration with the solution algorithm. The SIMPLE-algorithm which describes the fluid flow in the tube with linear tridiagonal systems of equations for the velocity, the pressure correction and the enthalpy, the changes of the state in the tube-collector-structure must be solved in a global system of equations for all tubes and collectors. By using the SIMPLE-algorithm we get a sparse coefficient matrix for the system of equations describing the pressure correction as well as the enthalpy. The system of equations representing the momentum balance is decoupled at the collectors, so the matrix will remain tridiagonal structured.

Finally results of a warm start-up of a HRSG are presented. The boiler was modelled in a tube-collector-structure; the calculation was realised using the SIMPLE-algorithm. Well-known phenomena which can be observed at real boilers could be verified by the simulation. Applying a ramp shaped heating, the circulation of the mass flow started in positive sense. If the heating rate exceeds a critical value, the circulation changes into negative direction. It was also possible to detect density wave oscillations at certain load conditions.

Our example shows, that the tube-collector-model represented in this paper is appropriate for the simulation of the fluid flow in a ramified tube network.

Acknowledgement: This study was supported by the Oesterreichische Nationalbank as Jubiläumsfondsprojekt Nr. 5040.

References

1. P. Nowotny, Ein Beitrag zur Strömungsberechnung in Rohrnetzwerken. Fortschritt - Bericht VDI, Series 6, Number 102, VDI Verlag, Düsseldorf 1982.

2. H. Rohse, Untersuchung der Vorgänge beim Übergang vom Umwälz- zum Zwangsdurchlaufbetrieb mit einer dynamischen Dampferzeugersimulation. Fortschritt - Bericht VDI, Series 6, Number 327, VDI Verlag, Düsseldorf 1995.

3. K. Ponweiser, W. Linzer, P. Szmolyan and E. B. Weinmüller, Dynamisches Verhalten von Naturumlauf-Dampferzeugern. Fortschritt - Bericht VDI, Series 6, Number 284, VDI Verlag, Düsseldorf 1993.

4. R. J. LeVeque, Numerical Methods for Conservations Laws: Computer Science And Applied Mathematics. Birkhäuser, Basel, 2. Edition 1992.

5. S. V. Patankar, Numerical Heat Transfer and Fluid Flow: Series in Computational Methods in Mechanics and Thermal Sciences. Hemisphere Publ. Corp., Washington, New York, London 1980.

6. K. Ponweiser, Numerische Simulation von dynamischen Strömungsvorgängen in netzwerkartigen Rohrstrukturen. Fortschritt - Bericht VDI, Series 6, Number 378, VDI Verlag, Düsseldorf 1997.

Modeling of Two Phase Flows in Modelica

Olaf Bauer[1] and Hubertus Tummescheit[2] (corresponding Author)
[1]Technical University of Hamburg-Harburg, email o.bauer@tu-harburg.de
[2]Lund University, Department of Automatic Control, SE-22100 Lund, Box 118, Sweden
email: Hubertus.Tummescheit@control.lth.se

Abstract. Modelica™[7] is an object-oriented language for modeling physical systems that was designed in the last years with the goal to become a standardized multi-domain modeling language. This paper describes a robust model for homogeneous and inhomogeneous two phase flows with dynamic or static slip correlation. It was developed in the context of developing a Modelica base library for thermo-hydraulic applications. The model describes the transient behavior of a fluid moving through a pipe during a phase change caused by heat transfer or pressure changes. Measurements from a refrigeration cycle were used to validate the model. Physical approaches were taken to model friction and momentum exchange between the phases. The model also includes the one-phase flow of liquid or vapor as limiting cases in order to make the simulation of a complete phase-transition possible. The model is numerically robust in all flow regions. Modelica's language features are used to structure the code for reusability in different contexts. These features make the model well suited for a reusable model library.

1 Introduction

In the past, detailed models of inhomogeneous two phase flows have only been available in specialized, domain-specific software systems like APROS [5] or SINDA/FLUINT [4]. These packages are well suited within their domain, but not for multi-domain models or for control design. Modelica overcomes these deficiencies. With its true equations and object-oriented language constructs it opens new horizons for physical modeling. The equations from the structured model are transformed into a ***hybrid differential-algebraic equation system*** (hybrid DAE). Hybrid means here, that both continuous time (the DAE) and discrete time dynamics can be present in the model, for details see [7]. Currently, the Modelica design group develops free basic libraries for various domains, e.g. for thermo-hydraulics. The simulation program DYMOLA [3] was used for the simulations.
Different models for transient two phase flows can be characterized by two main properties: homogeneous or inhomogeneous flow (referring to the flow speed of the phases) and equilibrium or non-equilibrium conditions of the thermodynamic state of the phases. Furthermore, there are many options for choosing the state variables, the discretization scheme and the boundary conditions. A thorough overview of many different models is presented in [6]. The model presented here can be parameterized as homogeneous or inhomogeneous. Both phases are always in thermodynamic equilibrium and it can have either a static or a dynamic slip-flow correlation. This range of models seemed best suited for the given purpose: transient models to evaluate control designs for refrigeration cycles. The test plant that was used for validation is sketched in Figure 1. Due to the horizontal evaporator pipe, it has almost separated flow conditions as in Figure 2, but the model can be used for any flow conditions.

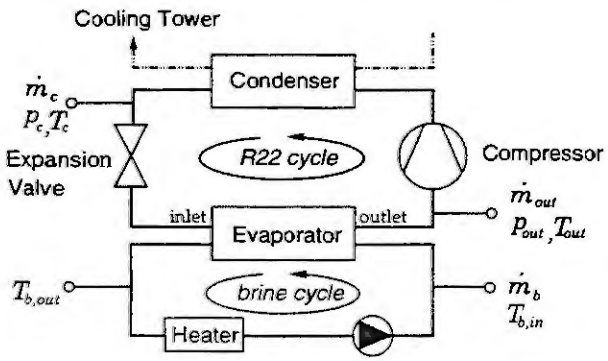


Figure 1: Refrigeration test plant

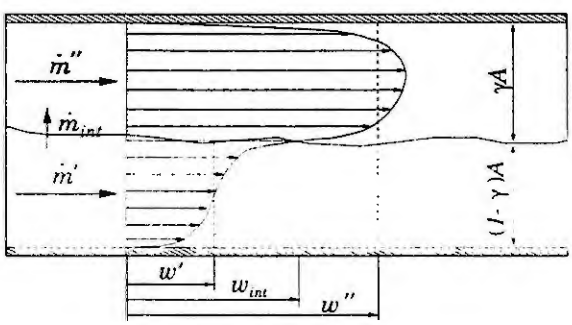


Figure 2: Separated Flow Conditions

2 Thermodynamic Model

The textbook form of the balance equations for mass and energy, [10], uses density ρ and internal energy u as states in the differential equations. For a constant control Volume V the equation can be written as:

$$\begin{pmatrix} \dot{m}_1 - \dot{m}_2 \\ \dot{H}_1 - \dot{H}_2 + \dot{Q} \end{pmatrix} = \frac{d}{dt}\begin{pmatrix} M \\ U \end{pmatrix} = V\begin{pmatrix} 1 & 0 \\ u & \rho \end{pmatrix}\frac{d}{dt}\begin{pmatrix} \rho \\ u \end{pmatrix} \tag{1}$$

The variable ρ is not suitable as a state variable in the liquid region, because the resulting system of equations is too stiff and thus unsuited for dynamic simulations. The differential of (ρ, u) can be rewritten in terms of the partial derivatives of (ρ, u) with respect to (p, h) as:

$$\frac{d}{dt}\begin{pmatrix} \rho \\ u \end{pmatrix} = \begin{pmatrix} \partial\rho/\partial p|_h & \partial\rho/\partial h|_p \\ \partial u/\partial p|_h & \partial u/\partial h|_p \end{pmatrix}\frac{d}{dt}\begin{pmatrix} p \\ h \end{pmatrix} \tag{2}$$

The partial derivatives of u follow from differentiation of $u = h - p/\rho$. Inserting Equation (2) into Equation (1) and then solving for $d/dt(p, h)$ gives:

$$V\left[\rho\frac{\partial\rho}{\partial p}\Big|_h + \frac{\partial\rho}{\partial h}\Big|_p\right]\frac{d}{dt}\begin{pmatrix} p \\ h \end{pmatrix} = \begin{pmatrix} \rho + h\,\partial\rho/\partial h|_p & -\partial\rho/\partial h|_p \\ 1 - h\,\partial\rho/\partial p|_h & \partial\rho/\partial p|_h \end{pmatrix}\begin{pmatrix} \dot{m}_1 - \dot{m}_2 \\ \dot{H}_1 - \dot{H}_2 + \dot{Q} \end{pmatrix} \tag{3}$$

This system is well suited for dynamic simulations. Modelica supports true equations and does not require the equations in an explicit form. The symbolic engine of DYMOLA will transform the equations to an explicit form when needed. Therefore, the equations are written in exactly this form in the Modelica code.

3 Equation of State

For efficient dynamic simulation of fluids it is necessary to have equations of state that are explicit in the state variables, p and h. The properties in the two-phase region can be expressed as a linear interpolation between dew- and boiling curve, which are both functions of p only. In the liquid region, linear Taylor expansions at the boiling curve for lines of constant enthalpies yield highly accurate results. In the vapor region, Taylor expansions at the dew curve for lines of constant pressure with varying order are applied. As an example, the following equations are obtained for the density:

$$\begin{array}{ll} \rho = \rho'(p'(h)) + \partial\rho/\partial p|'_h\,(p'(h))\,[p - p'(h)] & h < h'(p) \\ 1/\rho = v = xv''(p) + (1-x)v'(p) & h'(p) \le h \le h''(p) \\ 1/\rho = v = v''(p) + \partial v/\partial h|''_p\,(p)\,[h - h''(p)] + 0.5\,\partial^2 v/\partial h^2|''_p\,(p)\,[h - h''(p)]^2 & h > h''(p) \end{array} \tag{4}$$

where x is the quality $x = M''/M = (h - h'(p))/(h''(p) - h'(p))$. The approach requires functions for the properties and certain derivatives of saturated liquid and vapor. These functions are approximated by polynomials, except for $h'(p)$, which is the inverse function of $p'(h)$. This expression is required to maintain consistency of the equations.

$$p'(h) = \left[a + bh + ch^2\right]^6 \Leftrightarrow h'(p) = -0.5b/c - \left[(0.5b/c)^2 + (\sqrt[6]{p} - a)/c\right]^{0.5} \tag{5}$$

The approach gives a continuous transition on the phase boundaries and an increasing accuracy near the phase boundaries. It was applied to data of the refrigerant R22 on the basis of the NIST REFPROP database [8]. The relative error of $\rho(p, h)$ is shown in Figure 3. Similar equations are created for T, $\partial\rho/\partial p|_h$ and $\partial\rho/\partial h|_p$.

4 Hydrodynamic Model

There are three alternatives for the hydrodynamic model.All models are based on different equations for Δw. The *homogeneous* model assumes that $\Delta w = 0$. One *inhomogeneous* model uses an algebraic equation for Δw, giving a static slip-flow correlation, the other a differential equation for a dynamic slip-flow correlation. Thus, for the complete model, the state variables are pressure p, enthalpy h and mass flow $\dot{m}$ plus, for the dynamic slip-flow, the difference of the flow speeds Δw.

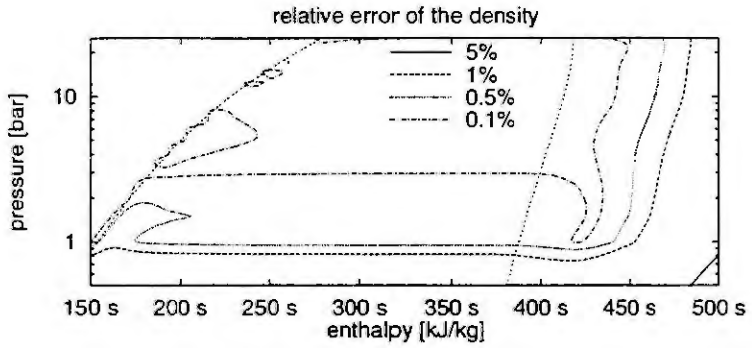


Figure 3: Level curves for the relative error of $\rho(p,h)$ for R22

A differential equation for the average mass flow rate in a pipe segment of length Δz follows from the overall momentum balance

$$\Delta z \cdot d\dot{m}/dt \approx dI/dt = \dot{I}_1 - \dot{I}_2 + \rho g \cos\varphi \Delta z A + (p_1 - p_2)A + F_{wall} \tag{6}$$

The wall friction force F_{wall} is commonly expressed with an empirical friction factor ξ

$$F_{wall} = -\frac{\xi}{8}\rho w|w|A_{wall} = -\xi\frac{\Delta z}{2D}\rho w|w|A = -\xi\frac{\Delta z}{2D}\dot{m}|w| \tag{7}$$

where $w := \dot{m}/(\rho A) = xw'' + (1-x)w'$. In two-phase flow it is often assumed that $\xi = \Phi_0^2 \xi'\rho/\rho'$, where Φ_0 is a modified two-phase multiplier and ξ' is the friction factor for a liquid flow with the same mass flow velocity $G = \rho w$. The momentum flow $\dot{I}$ can be expressed as.

$$\dot{I} = \dot{m}[\dot{x}w'' + (1-\dot{x})w'] = \dot{m}w + \dot{m}_{corr}\Delta w \qquad \text{with} \qquad \dot{m}_{corr} = (\dot{x} - x)\dot{m} = x(1-x)\rho\Delta w A \tag{8}$$

where $\dot{x} := \dot{m}''/\dot{m}$ is the flow quality, $\Delta w := w'' - w'$ is the difference between the average velocities and $\dot{m}_{corr}$ is the deviation of the vapor mass flow rate from the related value in a homogeneous flow with the same quality x. Viewed from an observer moving at average velocity w, the mass flow rate of the vapor equals $\dot{m}_{corr}$ while the liquid mass flow rate is $-\dot{m}_{corr}$. In the homogeneous case we have $\Delta w = 0$ and thus $\dot{m}_{corr} = 0$ and $x = \dot{x}$.
An equation for Δw is required for heterogeneous flow. An explicit algebraic equation follows from Levy's [12] equation for the flow quality $\dot{x}{=}\dot{x}(\gamma, \rho'/\rho'')$. Employing Equation (8) with $\dot{m} = \rho w A$ gives

$$\Delta w = w\frac{\dot{x} - x}{x(1-x)} = w\frac{\rho}{\rho'} \cdot \frac{\rho\sqrt{(1-2\gamma) + 2\gamma\rho'/\rho''} - \rho'}{2(1-\gamma)^2\rho' + \gamma\rho''(1-2\gamma)} \tag{9}$$

where $\gamma = V''/V = x\rho/\rho'' = (\rho - \rho')/(\rho'' - \rho')$ is the void fraction. A differential equation for Δw can be obtained from separate mass and momentum balances for the liquid and vapor phases [6]. Applied to a finite length Δz this differential equation becomes[1]

$$\Delta z \cdot d\Delta w/dt = \bar{w}_1\Delta w_1 - \bar{w}_2\Delta w_2 + [v'' - v'](p_1 - p_2) \tag{10}$$

$$+ \frac{\dot{m}_{int}(w_{int} - w'') + F''_{wall} + F''_{int}}{\gamma\rho' A} + \frac{\dot{m}_{int}(w_{int} - w') - F'_{wall} - F'_{int}}{(1-\gamma)\rho' A} \tag{11}$$

where $\bar{w} = 0.5(w'' + w')$ is the average velocity, $\dot{m}_{int}$ is the mass flow through the phase interface and w_{int} denotes the velocity at the interface, see Figure 2. For numerical robustness, denominator and numerator of the last two terms have to go to 0 at the same rate in the limits when $\gamma \to 0$ and $\gamma \to 1$. Simple approximations for w_{int} and the forces can be derived from this condition.
In spray flows, $\gamma \to 1$, the velocity on the surface of a small liquid drop hardly differs from the average velocity of the liquid inside the drop, thus $w_{int} \to w'$. Analogous considerations for small bubbles lead to $w_{int} \to w''$ for $\gamma \to 0$ and bubbly flows. Linear interpolation gives

$$w_{int} \approx \gamma w' + (1-\gamma)w'' \tag{12}$$

[1] Contrary to [6] the forces are counted positive in direction of the flow; the signs of F are thus reversed.

In most flow regimes the wall is completely wetted, but as the void fraction increases it gets partially dry, and for $\gamma \to 1$ the condition $F''_{wall} \to F_{wall}$ must be fulfilled. The fraction of the wall friction applied to the vapor should therefore rise from almost zero to one when the void fraction increases to one. This can qualitatively be achieved by using the quality as a weighting factor

$$F''_{wall} \approx x F_{wall} = \gamma \rho'' / \rho \cdot F_{wall} \qquad F'_{wall} \approx (1-x) F_{wall} = (1-\gamma)\rho' / \rho \cdot F_{wall} \tag{13}$$

Provided that the exchanges of mass between the phases are perpendicular to the flow direction, the equilibrium of tangential forces at the interface yields $F''_{int} + F'_{int} = 0$. The force is modeled analogous to the wall friction force Equation (7). If the velocity difference is positive, the vapor phase is decelerated, hence

$$F''_{int} = -F'_{int} = -\frac{\zeta}{8} \rho_{ref} \Delta w |\Delta w| A_{int} = -\zeta^* \frac{\Delta z}{2D} \rho_{ref} \Delta w |\Delta w| A = -\zeta^* \frac{\Delta z}{2D} \dot{m}_{ref} |\Delta w| \tag{14}$$

with $\zeta^* = \zeta A_{int} / A_{wall}$. The reference mass flow rate $\dot{m}_{ref}$ should be equal for both phases and be proportional to Δw. As explained above, this holds for the correction mass flow rate Equation (8), thus $\dot{m}_{ref} = \dot{m}_{corr}$ will be used. Inserting the above relations into Equation (11) yields

$$\Delta z \frac{d\Delta w}{dt} = \bar{w}_1 \Delta w_1 - \bar{w}_2 \Delta w_2 + [v'' - v'] \left[p_1 - p_2 - \Delta w \frac{\dot{m}_{int}}{A} \right] - \zeta^* \frac{\Delta z}{2D} \Delta w |\Delta w| \tag{15}$$

The interfacial mass flow rate is obtained from the vapor mass balance

$$\dot{m}_{int} = dM''/dt + \dot{m}''_2 - \dot{m}''_1 \tag{16}$$

where the unsteady term follows from differentiation of the condition $V = M''/\rho'' + M'/\rho'$ which for $V = \text{const}$ yields

$$(\rho' - \rho'') \frac{dM''}{dt} = -\rho'' \frac{dM}{dt} + V \left[\gamma \rho' \frac{d\rho''}{dp} + (1-\gamma) \rho'' \frac{d\rho'}{dp} \right] \frac{dp}{dt} \tag{17}$$

where $d\rho''/dp$ and $d\rho'/dp$ are the derivatives of the equilibrium vapor and liquid densities respectively; dM/dt and dp/dt follow from Equation (1) and Equation (3). The modified interfacial friction factor ζ^* depends on the interfacial friction and the ratio of the wall area and the surface area between the two phases. If measurement data is available, ζ^* can be fitted to that data. This is not an easy task, because friction factors and heat transfer coefficients have to be fitted simultaneously using nonlinear optimization techniques. Fortunately, even simple heuristics can result in satisfactory results, see Section 5. A more detailed derivation of the hydrodynamic model is available in [2].

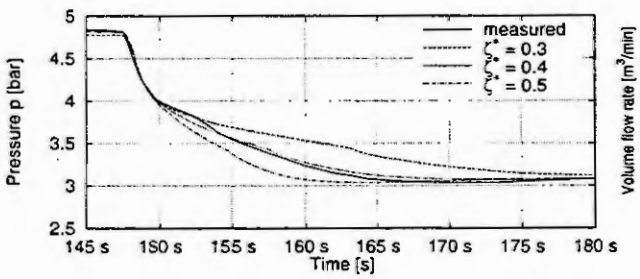


Figure 4: Simulated outlet pressure

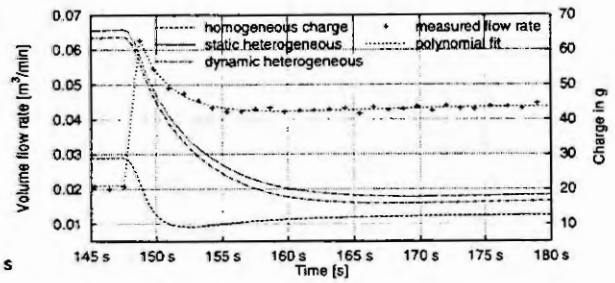


Figure 5: Charge of the Evaporator for a Step in the Outlet Volume Flow

Thermo- and hydrodynamic model form a system of coupled differential-algebraic equations with discrete variables for the phase. Following the principles of the finite volume method, the equations for each model are discretized to account for the spatial distribution of properties. The distinguishing feature of the finite volume method compared to a finite difference method is that the the balance equations are used in integral form. Thermo- and hydrodynamic equations are applied to different grid structures, that are staggered to each other by half a grid length, see Figure 6. The properties on the boundary of a pipe segment are approximated by the average properties of the cell located upstream. A thorough introduction to the finite volume method can be found in [9].

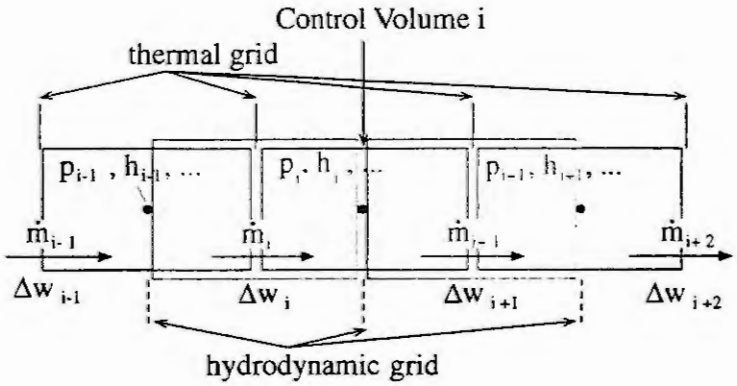


Figure 6: Staggered Grid in Finite Volume Method

5 Verification: Simulation of an Evaporator

The model is applied to simulate an evaporator in a refrigeration test plant at Danfoss, Denmark. Fig. 1 shows a simplified diagram of the refrigeration test plant and the data obtained from measurements performed by Antonius [1]. Complementary models are developed to compute the heat transfer through the pipe wall. For heat transfer coefficients and friction factors, empirical correlations are chosen that provide a continuous transition on the phase boundaries. Figures 5, 4, 7 and 8 show measured and simulated responses when there is an abrupt increase of the outlet volume flow rate, shown in Figure 5.
The homogeneous model for the mass flow rate and pressure agrees poorly with the measurements, but the heterogeneous model gives very good agreement with the measurements, see Figure 7 and Figure 8. The error of the simulated pressure is less than 5% for the heterogeneous and 35% for the homogeneous model. The improved accuracy is mainly a result of the larger evaporator charge, Figure 2, caused by the higher velocity of the vapor. The time it takes to evaporate the large amount of liquid causes a delayed decrease of the pressure.
For application of the dynamic equation for Δw, Equation (15), the interfacial friction factor ζ^* was set equal to ξ, which is a good approximation in the case of annular flows. It can be seen from the figures that the result is almost identical with the static equation and thus no further improvement is achieved. Figure 4 shows that a constant value $\zeta^* = 0.4$, also gives good results.
The temperature measurements taken at the plant where not suited for comparison of the dynamic tests because the thermo-elements used had too high time constants
All simulation results presented here are qualitatively and quantitatively very similar to the ones obtained by Antonius [1] with SINDA/FLUINT. The SINDA/FLUINT hydrodynamic model is based on two momentum balances for vapor and liquid. It is based on similar assumptions as the model with the dynamic slip-flow relation, but the actual form of the equations is quite different. The simulation times are about 10 times shorter in DYMOLA compared to SINDA/FLUINT, mainly due to the faster equations of state. In DYMOLA, a general purpose DAE solver was used, whereas SINDA/FLUINT uses specialized solvers for thermo-hydraulics which should be faster if all other conditions would be equal.

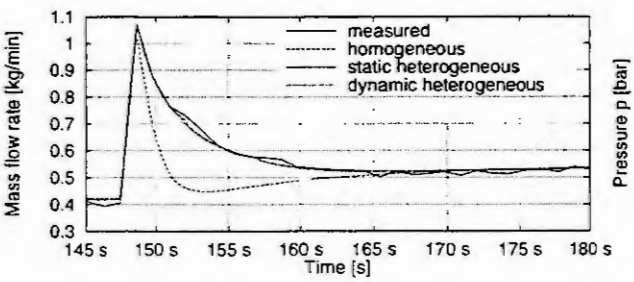


Figure 7: Simulated outlet mass flow rate

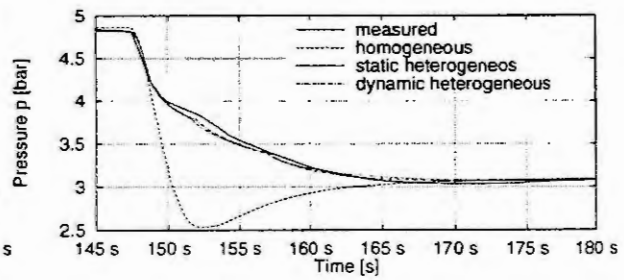


Figure 8: Simulated outlet pressure

6 Structure of the Modelica Code

Models that are developed for reusabilty in model libraries need a more careful design than models for one-time use. The design principles of the Modelica library for thermo-hydraulics are presented in [11]. The combination of *symbolic code manipulation* and high-level *submodel parameterization* makes it possible to choose between models of different complexity also from the graphical user interface. The submodels were assembled to the refrigeration test rig in Figure 1. Only the evaporator model is used for the verification of the model. The main components are the brine cycle side, the metal tube of the evaporator and the refrigerant side. Measurement data that is used as a boundary condition is encapsulated in a boundary model. This captures the decomposition into the actual physical objects. In thermodynamics it is useful to go one step further and decompose the system into conceptual subunits:

- The medium submodel – here for R22 – is a *replaceable* submodel.
- The *thermodynamic* model is separated from the *hydrodynamic* model.
- The type of the hydrodynamic model can be selected via a parameter.

This means that even the basic balance equations are contained in different submodels: the mass and energy balance are part of the thermodynamic model and the momentum balance is part of the hydrodynamic model. In Modelica it is possible to exchange *replaceable* models with other models that have the same model interface, i.e. the same *public* components. The medium model can be exchanged against any compatible model of a pure fluid, e.g. against the original medium property routine for R22 from NIST or against steam tables for water.

7 Conclusions

A flexible model for two-phase flows with different levels of modeling detail has been developed in the object-oriented modeling language Modelica. The results show clearly, that even for the simple models which are usually used in control design, a homogeneous flow assumption leads to wrong predictions of the predominant poles of the system. The main reason for this is, that the hydrodynamic models lead to different steady-state values for the mass of refrigerant in the evaporator. The different charge changes the thermal time constants for adjusting the mass in the evaporator through evaporation when the flow conditions change. That the resulting charge is the decisive factor can be concluded from the fact that the static slip-flow model performs equally well as the dynamic one (see Figure 5). For the given application the inhomogeneous flow model with static slip-flow relation seems to be the most appropriate. Nonetheless, the dynamic slip-flow model has several very interesting features: it is computationally faster, it is numerically robust because it has no singularities and it captures the dynamic effects of larger and faster transients better. Within the Modelica model library it could e.g. be reused for the dynamics of injecting liquids into gas flows.
The modeling language Modelica has proven to be very well suited for modeling this relatively detailed two-phase flow model. The model is modular and flexible. All three model variants can be selected via a parameter from DYMOLA's graphical user interface without re-compilation. The connector models are identical to Modelica's free thermo-hydraulic library[2] and thus the model can be mixed with lumped parameter models for valves and compressors. In this case, the model paradigm of using true equations for the model simplified the otherwise tedious translation from the mathematical model to simulation code enormously. The model also gives fast simulation in the DYMOLA environment.

Nomenclature

Symbol	Explanation	Symbol	Explanation
A	surface area	γ	void fraction
a, b, c	coefficients	ζ	interfacial friction factor
D	diameter	ζ^*	$\zeta A_{int}/A_{wall}$
F	axial force	ξ	friction factor
g	acceleration due to gravity	ρ	density
h	specific enthalpy	φ	pipe inclination
$\dot{H}$	enthalpy flux	Φ	two-phase multiplier

[2]The Modelica base library for thermo-hydraulics is currently under development at Lund University by the second author.

I	axial momentum	$_{1,2}$	inlet,outlet
$\dot{I}$	axial momentum flux	$_{c}$	condenser
$\dot{m}$	mass flow rate	$_{corr}$	correction
M	mass	$_{b}$	brine
p	pressure	$_{h}$	h = const.
$\dot{Q}$	heat flux	$_{in}$	inlet
t	time	$_{int}$	phase interface
T	temperature	$_{out}$	outlet
u	specific internal energy	$_{p}$	p = const.
U	internal energy	$_{ref}$	reference
v	specific volume	$_{wall}$	wall
V	volume	$'$	liquid
w	axial velocity	$''$	vapour
x	quality	Δ	difference
$\dot{x}$	flow quality	d	total differential
z	axial position	$\partial x/\partial y\|_z$	$\partial x/\partial y$ at constant z

References

[1] ANTONIUS, J., Distribuerede fordampermodeller på flere detaljeringsniveauer Danmarks Tekniske Universitet, 1998

[2] BAUER, O. Modelling of Two-Phase Flows in Modelica™, Technical Report LUTFD2/TFRT–5629–SE, Department of Automatic Control, Lund Institute of Technology, 1999.

[3] ELMQVIST, H. AND D. BRÜCK AND M. OTTER Dymola — User's Manual, Dynasim AB, Research Park Ideon, Lund, Sweden, 1999.

[4] CULLIMORE, B., S. G. RING, R. G. GOBLE AND C. L. JENSEN Sinda/Fluint Users Manual for Version 3.2, Cullimore and Ring Technologies, Inc. Littleton, Colorado, 1996.

[5] JUSLIN, K. Experience on mechanistic modelling of industrial processes with APROS, Mathametics and Computers in Simulation, 39:505–511, 1995.

[6] KOLEV, N. I., Transiente Zweiphasenströmung, Springer-Verlag, Berlin, Heidelberg, 1986.

[7] MODELICA DESIGN GROUP The Modelica Language Specification, Version 1.2, Homepage: `http://www.modelica.org/`, 1999.

[8] NIST REFPROP DATABASE National Institute of Standards and Technology, Version 6.0.

[9] PATANKAR, S. V., Numerical Heat Transfer and Fluid Flow Hemisphere Publishing Corp., 1980.

[10] STEPHAN, K.; BAEHR, H. D., Wärme und Stoffübertragung, 3. Auflage, Springer-Verlag, Berlin, Heidelberg, 1998.

[11] TUMMESCHEIT, H., Objektorientierte Modellierung Physikalischer Systeme, Teil 11, *to appear in* Automatisierungstechnik 48, March 2000.

[12] WANG, H. Modelling of a Refrigerating System Coupled with a Refrigerated Room, Delft University of Technology, 1991.

INTERFACIAL NON-TURBULENT THICKNESS IN AGITATED SYSTEMS

Winston Khan
U.P.R. Mayagüez, Puerto Rico
Tel. (787) 265-3844, Fax: (787) 832-1135

Abstract. On the approach of turbulent vortices or eddies to an interface or free surface under varying interfacial conditions, it is believed and evidenced by various observations and experiments based on agitated systems, stirred or otherwise, that the scale of turbulence changes at some depth beneath the surface. This depth is assumed to be proportional to the non-turbulent thickness which varies with contaminated interfaces of soluble or insoluble films. Consequently, the frequency with which eddies are impelled into the interface from the bulk liquid depends on this non-turbulent thickness, which is a function of the elastic properties of the film and the turbulent parameters. Most interfacial transport phenomena depend on this eddy frquency, and the interaction of the penetrating eddies with the free surface is germane to naval interest, as this interaction can determine the hydrodynamic signature of surface ships. It would appear, that the importance of this non-turbulent thickness layer, cannot be over emphasized and that its determination is worthy of investigation.

Introduction.

Surface or interfacial phenomena have gained paramount importance in industry, and in particular, the shipping industry, which lends itself to naval interest spanning the last decade. It is in this light, that vortex, eddy and jet interaction with clean and contaminated surfaces are being investigated.

The scale change depth, which is related to the non-turbulent thickness, hence the eddy frequency penetration, is germane to all industrial enterprises involving these physical situations.

The pioneering work of Danckwerts in determining mass-transfer across turbulent interfaces produced a formula involving the eddy frequency penetration which depended on the non-turbulent thickness layer beneath the surface. Hence, the effect of the interfacial conditions on the eddy frequency penetration translates to the effect on the non-turbulent thickness that is being investigated.

In the circumstances, a quick review of the salient aspects of Danckwerts theory would not only be revealing but inevitable. Danckwerts defined a surface age distribution funcition, $\phi(\theta)$, where θ is the time of exposure of an element of surface to the gas or its age. On this basis, he formed the relation that $\phi(\theta)\, d\theta = \phi(\theta - d\theta)d\theta(1 - sd\theta)$, therefore,

$$\phi(\theta) = \phi(\theta) - \frac{d\phi}{d\theta}d\theta - s\phi(\theta)d\theta, hence \frac{d\phi}{d\theta} = -s\phi \text{ , giving } \phi = se^{-s\theta} \text{ , since } \int_o^\infty \phi d\theta = 1 \text{, for}$$

a unit surface area of exposure, and, where s is the fractional rate of replacement of elements belonging to any age group, which is interpreted physically to mean the eddy penetration frequency into the interface from the bulk phase. The rate of absorption of gas or mass transfer was given by $R = (c^* - c_\circ)\sqrt{D}\int_o^\infty se^{-s\theta}\frac{1}{\sqrt{\pi\theta}}d\theta$, and using the La place transform, we get, $R = (c^* - c_\circ)\sqrt{Ds}$

In fact, most interfacial phenomena are related to s which is the eddy penetration frequency into the interface from the bulk phase.

Danckwerts obtained this result for clean interfaces, but the theoretical treatment does not preclude contaminated interfaces, which have been supported experimentally for both clean and films of insoluble contamination.

Method.

From the foregoing, we should endeavor to determine the effect of contaminated surfaces on the eddy penetration frequency, which is related to the non-turbulent thickness layer under all interfacial conditions, including the most general case of soluble films. When a liquid is set into turbulent motion, it becomes a mass of eddies of varying sizes and frequencies. If we assume that the size of an eddy given by λ is comparable to the wavelength over the various wavenumbers that characterize turbulence, then we may write that $S=\frac{Vo}{\lambda}$, where Vo is a characteristic turbulent velocity with which an eddy is impelled into the interface. As the free surface is approached by an eddy of liquid, fluctuations normal to the surface are diminished in comparison to those parallel to the surface, which are accentuated, resulting in an interfacial turn around or rebound of the approaching eddy. The depth at which this rebound takes place corresponds to the depth at which the dynamic properties begin to influence or impact the interface, which then retaliates through its elastic properties to negate the dynamic pressures exerted by the approaching eddy. We assume that this depth is comparable to the eddy size migrating into the interface. The eddy rebound would imply an eddy penetration frequency damping, and, at the same time, a redistribution of the energy, hence, a clue to the model involved.

The model is that of a body of fluid in the form of an eddy or turbulent vortex which impinges on the interface under different interfacial conditions of clean or contaminated with insoluble and soluble films.

If λ is the eddy size, Vo the characteristic turbulent velocity, γ the surface tension and C_s^{-1}, the surface compressional modulus of elasticity, then energy considerations show that,

$$\rho V_0^2 \lambda^3 \alpha (\gamma + C_s^{-1}) \lambda^2,$$

Hence $\lambda \alpha \frac{\gamma + C_s^{-1}}{\rho V_0^2}$. We interpret the eddy size λ to be comparable to the depth at which the dynamics of the eddy interacts with the interfacial conditions and synonymous with the non-turbulent thickness layer. The frequency $S=\frac{Vo}{\lambda} \alpha \frac{\rho v_0^3}{\gamma + C_s^{-1}}$, which agrees with experimental observations for both clean and contaminated interfaces of insoluble films. The problem now resolves itself into extending the model to the case of soluble interfacial contamination. In this case the surface compressional modulus of elasticity, C_s^{-1}, would become a function of time. In the circumstances, the formula for λ which is synonymous with the non-turbulent thickness layer beneath the interface must incorporate this solubility effect through the surface compressional modulus of elasticity C^{-1}_s

The formula suggested, is that of $\lambda = \frac{\gamma + \gamma_0 N_0 e^{-s t}}{\rho v_0^3}$, where $C_s^{-1} = C_{0,S}^{-1} e^{-st} = \gamma_0 N_0 e^{-st}$. This formula obeys all the boundary conditions and reduces to the special cases of 1) Clean interface, 2) Insoluble films.

The formula for λ should be considered as a semi-empirical one, as it is obtained indirectly from the boundary conditions and the observed hyperbolic and exponential forms that manifest themselves in this kind of problem. It was impossible to satisfy the boundary conditions with a hyperbolic function for C_s^{-1}, hence the chosen exponential form.

Perhaps, the behavior of surfactants and their elastic effects should be examined at this stage. It is well known that variations in area in general will induce variations in surface tension, such that, $\frac{d\gamma}{\gamma_0} = N \frac{dA}{A}$, where A is the available area per molecule of surfactant and gives rise to the compressional

modulus of elasticity, $C_{s,0}^{-1}$ defined as $N\gamma_0$, where N is some positive number and γ_0 the surface tension for a clean interface. Also, $\gamma = \gamma_0 - \pi$, where π is the back spreading pressure of the film molecules, from which we obtain, $A\frac{d\gamma}{dA} = N\gamma_0 = C_{s,0}^{-1} = A\frac{d\pi}{dA}$.

Conclusions.

The formula, $\lambda = \frac{\gamma + C_{0,s}\, e^{-st}}{\rho V_0^{\,3}}$, reduces to the special cases of clean and contaminated interfaces with insoluble films, which have been corroborated experimentally. It also satisfies all boundary conditions for all cases. However, in the case of soluble films, the most general case, the value of $\lambda \rightarrow$ clean interface as $t \rightarrow \infty$, but this would not obtain, as a certain equilibrium state would be achieved after some time, when the amount of soluble film molecules into the bulk is exactly balanced by the amount of film molecules returned by the eddies impelled into the interface. In short, a state where absorption is balanced by desorption.

References.

1. Anthony, D. G. On the interaction of a submerged jet with clean or contaminated free surfaces. Phys. Of fluids A, Vol. 3, No. 2. Feb. 1991
2. Walker, D. J. Interaction of a turbulent jet with a free surface, Office of Naval Research, Workshop, March, 1992.
3. Davies, J. T. Turbulence Phenomena, Academ Press, 1972
4. Levich, J. G. Physicochemical Hydrodynamics, Prentice Hall, 1962
5. Danckwerts. P.V. Significance of Piquid film coefficients in gas absorption, Ind. Eng. Chem. 43, 1960 (1951)
6. Khan, W. Eddy damping by surfactants, Int. jour mat. mod. Vol. 5, no. 6, (1984)
7. Fish, S. Vortex dynamics in the presence of free surface waves, Phys. Fluids A3, (4), April, 1991
8. Lagory Larry, M. Interaction of Wake turbulence with a free surface, Phys. Fluids 8 (3), March, 1996
9. Walker, D. J. Froude Number effects in free surface jets. J. Fluid Mech. 243, 699-720, 1992

Dynamical Physical Modeling of a Supercharged Internal Combustion Engine

P. Skorjanz[1], R. Korb[1], S. Jakubek[1], B. Lutz[2]
[1] Vienna University of Technology
Gußhausstrasse 27-29, A-1040 Wien
skorjanz@impa.tuwien.ac.at
[2] Jenbacher Energiesysteme AG
A-6200 Jenbach, Austria
b.lutz@jenbacher.com

Abstract. The paper describes dynamical physical modeling of a large stationary supercharged 4-stroke spark ignition engine with gas mixer and throttle-plate, that is primarily used to drive a threephase current generator for production of electrical energy for an grid (in parallel or isolated mode). By means of the mean-value modeling technique an engine model is achieved, which is sufficiently accurate for control design purposes. The adaptation of the model parameters is based on a-priory knowledge of single components and on measurement data. Agreement between measured data and simulated results is shown to be good.

Introduction

The objective of the presented work is to provide an engine model sufficiently accurate for control design studies. Depending on the engine operation mode the speed of the engine is fixed and the produced power is an output variable (parallel mode) or the speed is free to vary and the produced power has to be treated as a load (isolated mode). The aim of speed-control in isolated operation, which should be improved, consists in keeping the speed of the engine as close as possible to the set point corresponding to the mains frequency while meeting (stationary) emission limits. The model will be used to study alternative control concepts that employ present control equipment and to study effects of changes in type and position of the control equipment. The valid operating region of the model covers the entire power range and a limited band of speed values around the set point under normal operation conditions.
The kind of modeling approach has to be selected according to the main objective. While black-box models are useful for controller parameter design, their main drawbacks are a restricted region of validity if the system changes and lack of information about the cause of the system behaviour. Since a model based on physical relationships provides significantly more information about the real system and is more likely to be able to model changes in the systems structure in a correct way, this kind of modeling technique is chosen.

The model design has to take into account that the computational effort for simulation must not be too high. A number of simplifications have to be made, while maintaining the simulation accuracy at a high enough level. In literature different modeling approaches are presented, varying in accuracy and computational effort. Besides approaches using methods of higher dimensional computational fluid dynamics, which lead to partial differential equations and are computationally much too expensive, there exist a number of simpler approaches, assuming the system can be described using differential equations with lumped parameters. Many papers have been published on this topic, [1-6] are examples. Further simplification is achieved by assuming that on account of the engine design (many cylinders) and the relevant time scale for speed- and power-control the discontinuous working method can be considered as continuous and fast transient changes in some variables can be considered as instantaneous. In this way the method of mean value modeling is introduced [1-4,6].

For the given modeling purposes mean value modeling is the proper choice. Mean value models have been presented for supercharged Diesel engines and non-charged spark ignition engines with fuel injection. Some extensions have to be made for being able to model an engine with gas mixer, turbocharger and throttle plate.

Overview of submodels and interactions

An overview of the engine's components is shown in figure 1. The parts of the engine are described based on physical relationships, that sometimes require fitting of parameters.

A classification of the components can be given, distinguishing between components containing states and components acting instantaneously. The main components containing states are the flow path of the fluid, the mechanical part of the engine including piston, crankshaft, flywheel and generator, the rotor of the turbocharger and the mechanical part of the actuator and throttle plate. The rate of change of the states is calculated according to physical relationships for instantaneously acting components. Essential parts to be modeled acting

instantaneously include air filter, gas mixer, compressor, intercooler, throttle plate (as part of the flow path), cylinders, turbine and losses in the exhaust section.

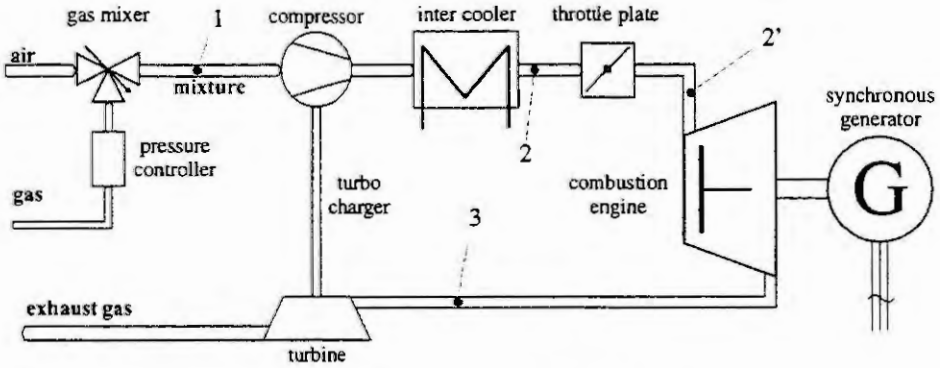


Fig. 1: Main components of the engine

Parts of the flow path of the fluid with almost the same value of pressure in the relevant time scale are combined to ideally mixed containers. The volume of these containers is calculated corresponding to the design. The values of the container state variables, namely pressure, temperature and air-fuel ratio, result from mass and energy conservation laws. Four containers are introduced (in figure 1 numbered 1, 2, 2', 3), modeling the volumes in front of the gas mixer (1), between compressor and throttle plate (2), between throttle plate and inlet valves (2') and between outlet valves and turbine (3). The main equations are (for each container i):

$$\dot{m}_i = \dot{m}_{in} - \dot{m}_{out}$$

$$p_i^t = \frac{m_i}{V_i} R\, T_i^t$$

$$\dot{T}_i^t = \frac{1}{c_v\, m_i}\left(\dot{m}_{in}\left(c_p\, T_{in}^t - c_v\, T_i^t\right) - \dot{m}_{out}\, R\, T_i^t + \dot{Q}_i\right)$$

$$\dot{c}_{air,i} = \frac{\dot{m}_{in}}{m_i}\left(c_{air,in} - c_{air,i}\right)$$

Changes in engine and turbocharger speed are calculated according to the momentum equation. The dynamics of the throttle plate are approximated by a simple second order transfer function model:

$$I_{engine}\, \dot{\omega}_{engine}\, \omega_{engine} = P_{produced}(\dot{m}_{engine}, \omega_{engine}, c_{air,engine}) - P_{external}$$

$$I_{turbocharger}\, \dot{\omega}_{turbocharger}\, \omega_{turbocharger} = P_{turbine}(p_3^t, T_3^t, \omega_{turbocharger}) - P_{compressor}(p_1^t, p_2^t, T_1^t, \omega_{turbocharger}) - P_{friction}(\omega_{turbocharger})$$

$$\ddot{\varphi}_{throttle} + 2\,\omega_n\,\zeta\,\dot{\varphi}_{throttle} + \omega_n^2\,\varphi_{throttle} = \omega_n^2\,\varphi_{desired}$$

Thus the model comprises states for pressures, air-fuel ratios and temperatures for the containers, engine and turbocharger speed and throttle plate position. Mass flows into and out of the containers are calculated according to adjacent pressures by the use of submodels for compressor and turbine, throttle plate, cylinders and losses in the air filter. These submodels mainly base on physical relationships including parameters to be fitted.

$$\dot{m}_{compressor} = f_{compressor}(\omega_{turbocharger}, p_1^t, T_1^t, \frac{p_2^t}{p_1^t})$$

$$\dot{m}_{turbine} = f_{turbine}(\omega_{turbocharger}, p_3^t, T_3^t)$$

$$\dot{m}_{throttle} = \mu A \frac{p_2^t}{\sqrt{R\,T_2^t}}\beta(\Pi) \text{ with } \Pi = \frac{p_2^t}{p_{2'}},\ \beta = \begin{cases} \sqrt{\frac{2\kappa}{\kappa-1}}\sqrt{\Pi^{-\frac{2}{\kappa}} - \Pi^{-\frac{\kappa+1}{\kappa}}} & \text{for } \Pi < \left(\frac{\kappa+1}{2}\right)^{\frac{\kappa}{\kappa-1}} \\ \sqrt{\kappa\left(\frac{2}{\kappa+1}\right)^{\frac{\kappa+1}{\kappa-1}}} & \text{for } \Pi \geq \left(\frac{\kappa+1}{2}\right)^{\frac{\kappa}{\kappa-1}} \end{cases}$$

$$\dot{m}_{engine} = V_{displacement}\, \pi\, \omega_{engine} \frac{p_{2'}}{R\, T_{2'}} \eta_{volumetric}(p_{2'})$$

$$\dot{m}_{airfilter} = f_{airfilter}(p_1)$$

$$c_{air,gasmixer} = f_{gasmixer}(\dot{m}_{gasmixer}, \alpha_{gasmixer})$$

The air-fuel ratio arises from a submodel of the gas mixer and is transmitted to the containers by mass flows. The combustion process is considered to be quasi-continuous. Mass flow to the cylinders, air-fuel ratio and the value of the effective efficiency of the engine influence the power produced by the engine. The produces power can either be transfered to an electric grid that keeps the engine speed on a certain value or which can be used for acceleration or deceleration (in combination with a load) when operating in isolated mode. The power of compressor and turbine acting on the rotor of the turbocharger can be calculated by use of physical relationships depending on the values of the turbocharger speed, adjacent pressures and temperatures and given data-sheets for characteristic parameters.

Parametrization of subsystems

Fitting of previously unknown characteristic curves and input-output maps of subsystems is carried out on basis of measurement data that is recorded in parallel operation mode, as the open loop system is locally stable for this kind of operation in opposition to the behaviour in isolated mode, where it is locally unstable. The excitation is realized by variations of the input variables throttle plate position and gas mixer position. According to the kind of operation when the measurements for model parametrization are recorded there is no speed-dependency of the parameters. But as the speed-range of interest covers only a limited band around the engine speed for which the data is recorded, one can expect the model to be sufficiently accurate when the effects of changes in engine speed are properly modeled in the physical relationships which are influenced by varying engine speed.

The mass flow through the engine could not be measured directly, but had to be determined by measured fuel consumption and measured air-fuel ratio. As the mass flow plays a significant role for the fitting of parameters, possible measurement errors in the underlying variables have to be avoided or at least detected. The values of calculated mass flow can be tested by use of prior knowledge about the system components compressor and turbine.

Based on measurement data the following characteristics and input-output maps are fitted:

- Input-output map of the effective efficiency of the engine dependent on mixture density and air-fuel ratio
- Characteristic of the volumetric efficiency of the engine dependent on mixture density
- Input-output map of the amount of gas added in the gas mixer dependent on the mass flow through the gas mixer and the gas mixer position
- Characteristic for the effective opening area of the throttle plate dependent on the throttle plate position
- Input-output map for the exhaust gas temperature dependent on mass flow and air-fuel ratio
- Characteristic for losses in the air filter, the intercooler and the exhaust section dependent on mass flow

The selection of the input variables for the characteristics and input-output maps is based on physical and practical considerations.

The following input-output maps depend on informations provided by the manufacturer:

- Input-output map for efficiency and volumetric flow through the compressor dependent on pressure ratio and turbocharger speed
- Input-output map for efficiency and mass flow through the turbine dependent on pressure ratio and turbocharger speed

For the model to be able to simulate the behaviour of the engine as realisticly as possible, input-output maps are not realized by means of simple look-up tables, avoiding problems with accuracy and inconveniently spaced input points. Depending on the task special functions with fitting parameters, polynomials (of appropriate dimension), splines and combinations of these approaches are used. By selecting the right kind of fitting function, good interpolation and respectable extrapolation properties are achieved.

Results

The simulation model of the entire engine is realized in MATLAB/Simulink. The behaviour of submodels fits measurement data in an excellent way. The model in isolated operation can only be compared when the engine speed is controlled, as the open loop system is locally unstable for this kind of operation. As shown in figure 2 for the engine speed and the pressure after the throttle plate, the approximation of the significant states of the engine is qualitatively and quantitatively good. During measurement, the load was increased in steps of 50 kW. In some operating regions the controller of the real system doesn't perform well (time 70 – 120 seconds). The controlled model reproduces the same effect.

The model helps to understand the operating behaviour of the engine and is well designed to analyze the control of engine speed and the effect of modifications in control equipment. With some modifications a linearized model can easily be derived from the Simulink-model. As the model is built in a modular way, effects of changes in the system structure can be studied with little modifications.

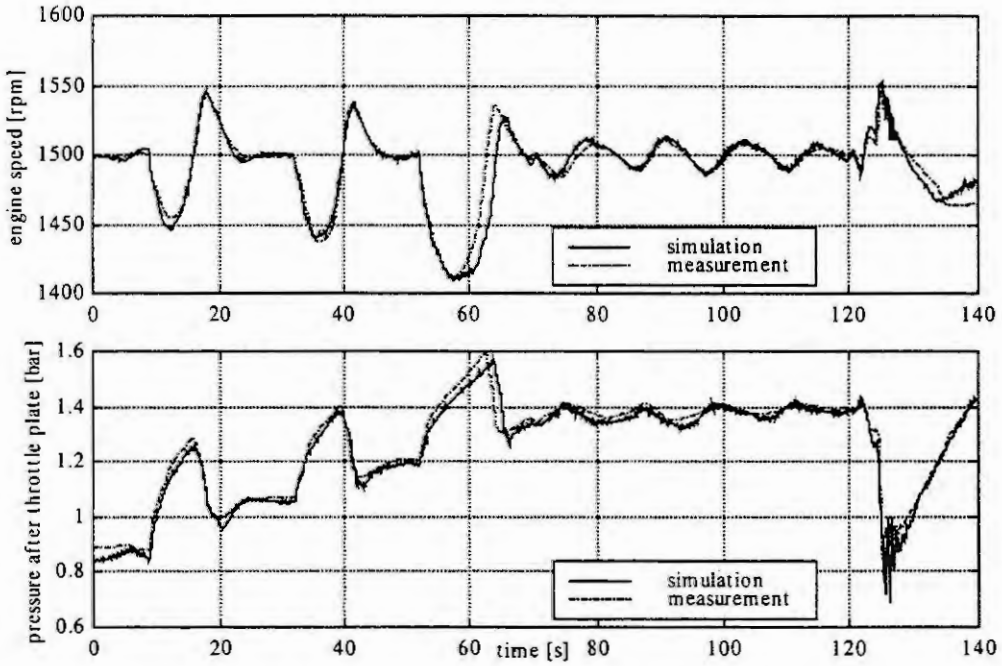


Fig. 2: Comparison of engine measurement and simulation data for operation as an isolated system

Summary

The concept of mean-value modeling was shown to work also for the special case of a stationary supercharged spark ignition engine with gas mixer and throttle plate. The accuracy as well as the simulation speed of the derived engine model based on physical relationships are well suited for control studies. As the model structure is modular, it can easily be modified to study e.g. the effects of different actuators. The act of modeling itself helps to understand the engines operating behaviour. A number of known and unknown parameters has to be included in the model in form of input-output maps, which are stored in form of fitting functions. For good inter- and extrapolation properties, the fitting functions for input-output maps have to be selected with care.

References

1. Kao, M., Moskwa, J. J., Turbocharged Diesel Engine Modeling for nonlinear Engine Control and State Estimation. Transactions of the ASME, Vol. 117, March 1995, 21 - 30

2. Dobner, D. J., A Mathematical Engine Model for Development of Dynamic Engine Control. In: Congress and Exposition, Cobo Hall, Detroit, February 25 – 29, 1980

3. Hendricks, E., Sorenson, S. C., Mean Value Modelling of Spark Ignition Engines. In: International Congress and Exposition, Detroit, Michigan, February 26 – March 2, 1990

4. Schmidt, Ch., Kessel, J.-A., Isermann, R., Modellbildung und adaptive dynamische Steuerung von Dieselmotoren, Abschlußbericht. Forschungsvereinigung Verbrennungskraftmaschinen e.V., Heft 602, 1996

5. Moskwa, J. J., Hedrick, J. K, Modeling and Validation of Automotive Engines for Control Algorithm Development, ASME J. of Dynamic Systems, Measurement and Control, June 1992

6. Jakubek, S., Jörgl, H.P., Korb, R., Physikalische Modellierung eines Gasottomotors für regelungstechnische Zwecke. Diplomarbeit an der TU Wien, Oktober 1997

MODELLING OF THE IRON LOSSES IN LAMINATED MAGNETIC MATERIALS USING A DYNAMIC PREISACH MODEL

L. Dupré[1], R. Van Keer[2], J. Melkebeek[1]
[1] Department of Electrical Engineering, University of Gent
St. Pietersnieuwstraat 41, B-9000 Gent, Belgium
[2] Department of Mathematical Analysis, University of Gent
Galglaan , B-9000 Gent, Belgium

Abstract. In this paper we report on recent advances in the modelling of magnetic losses in steel laminations used in electromagnetic devices. The integrated lamination moving dynamic Preisach model, used to evaluate the dynamic magnetisation loops under distorted unidirectional flux patterns, is described. The main goal is to compare and to describe the correlation between the advanced model based on the Preisach theory and a classical model based on the statistical loss theory. The two models are validated by the comparison of numerical experiments and measurements on two different materials.

Introduction

A thorough modelling of the macroscopic magnetic behaviour of laminated materials requires that the material characteristics take into account the presence of non-linear and hysteretic effects. Moreover, the macroscopic eddy currents inside the lamination, significantly altering the flux distribution which is then no longer uniform, must be taken into account. These two requirements must be fulfilled together because the non-linearity involved in the phenomena does not allow a separate solution of the two problems. A possible approach to the analysis of the phenomena inside the ferromagnetic laminations is to describe the interacting hysteresis and eddy current effects in terms of the macroscopic fields. This is performed by means of numerical methods for the solution of Maxwell equations in magnetic cores combined with advanced hysteresis models like the Preisach model [1]. It is well known in the literature [2] that the classical Preisach model implies some hysteresis properties, such as the congruency property, which are not necessarily met by all materials. Then, a further improvement can be obtained if more sophisticated models such as the product model [3] and the moving model [4] are used. Notice that the output of all these models are excitation rate independent. A general approach to the calculation of iron losses in soft magnetic laminated materials under unidirectional flux $\varphi(t)$ is based on the separation of losses into three components: the hysteresis losses P_h, the classical losses P_c and the excess losses P_e, $[Watt/m^3]$ see [5], [6]. In order to be able to model the excess losses by the Preisach theory, the hysteresis effects must be defined in a frequency dependent way [7]. Therefore, in [8], [9] a rate-dependent Preisach model was introduced which takes into account such a frequency dependence.
Fortunately, even if the handling of the material characteristic is rather difficult, the geometrical domain of the problem is, in a first approximation, one-dimensional [10] because the thickness of the lamination is much smaller than its other dimensions.

Statistical loss theory

According to the statistical loss theory [5], [6] the total loss P_t under sinusoidal, unidirectional flux φ with frequency f is given by:

$$P_t = P_h + P_c + P_e \equiv fW_h(B_m) + \frac{1}{6}\sigma\pi^2 d^2 f^2 B_m^2 + P_e \tag{1}$$

Here, the magnetisation process in the cross section S, perpendicular to the magnetic flux in the magnetic lamination of thickness d, can be described in terms of n simultaneously active correlation regions. For several alloys, n is a linear function of the excess field $H_{exc} = P_e/(4fB_m)$, i.e.

$$n = n_0 + H_{exc}/V_0 \tag{2}$$

When Eqn.(2) holds, the excess losses under a sinusoidal flux excitation with frequency f and maximum induction $B_m = \varphi_{max}/S$, can be written as

$$P_e = 2B_m f(\sqrt{n_0^2 V_0^2 + 16\sigma G S V_0 B_m f} - n_0 V_0) \tag{3}$$

Here, σ is the electrical conductivity and G=0.1357 is a dimensionless coefficient due to eddy current damping. V_0 and n_0 are fitting parameters describing microstructural features like grainsize and crystallographic texture [11]. The physical considerations, underlying (1) and (3), reveal that it is possible to estimate not only the average loss per cycle, but also the instantaneous loss $P_t(t)$ at time t, in a lamination. Under the condition of negligible skin effects, it may be written as

$$P_t(t) \equiv P_h(t) + P_c(t) + P_e(t) \equiv \frac{dB_a}{dt}(H_h(t) + H_c(t) + H_e(t)) \equiv \frac{dB_a}{dt}H_s(t) \tag{4}$$

with

$$H_c = \frac{1}{12}\sigma d^2 \frac{dB_a}{dt}, H_e = \frac{n_0 V_0}{2}\left(\sqrt{1 + \frac{4\sigma G S}{n_0^2 V_0}\left|\frac{dB_a}{dt}\right|} - 1\right) sign\left(\frac{dB_a}{dt}\right), B_a = \frac{\varphi}{S} \tag{5}$$

where the hysteresis field H_h of course depends on the instantaneous induction B_a and where the classical field H_c and the excess field H_e depend on dB_a/dt and where H_e is allowed to vary also with B_a through n_0 and V_0.

Preisach modelling

The scalar, rate independent Preisach model provides quite an accurate description of hysteresis effects in magnetic materials. In this model, each Preisach dipole has a non symmetric hysteresis loop defined by the two switching fields α and β ($\beta \leq \alpha$) and the parameter k_n, see Fig.1. Depending on the history of the effective magnetic field H_e, the magnetisation ϕ of the dipole takes a value within the interval [−1,+1]. In the rate-dependent Preisach model (DPM), as described in [12], the switching rate of each dipole is given by:

$$\frac{d\phi}{dt} = \begin{cases} k_d(H_e(t) - k_n\phi - \alpha) & , H_e(t) > \alpha + k_n\phi \text{ and } \phi < +1 \\ k_d(H_e(t) - k_n\phi - \beta) & , H_e(t) < \beta + k_n\phi \text{ and } \phi > -1 \\ 0 & , \text{in the other cases} \end{cases} \tag{6}$$

Then, the magnetisation $M(t)$ is obtained from

$$M(t) = M_{rev}(H_e(t)) + \frac{1}{2}\int_{-\infty}^{\infty} d\alpha \int_{-\infty}^{\alpha} d\beta \phi(\alpha, \beta, t) P(\alpha, \beta) = B(t) - \mu_0 H(t) \tag{7}$$

where $P(\alpha, \beta)$ is the Preisach distribution function (PDF) and the effective field $H_e(t)$ is obtained from the applied field $H(t)$ and the corresponding magnetisation $M(t)$, viz $H_e(t) = H(t) + k_m M(t)$. In this model, the (α,β)-halfplane is divided into 3 subregions, i.e. two regions in which ϕ equals *+1* and *-1* respectively, and a third region, in which the dipoles are in an intermediate state ($-1 < \phi < +1$). The shape of the third region is defined by the material parameter k_n and by the time variation of the magnetic field $H_e(t)$. It is shown in [12] that the dynamic hysteresis losses obtained by this DPM, described by (6)-(7), follow the frequency dependence of (3).

The identification of the PDF $P(\alpha, \beta)$ can be performed in different ways. The procedure proposed by Mayergoyz [2], which is based on the experimental evaluation of numerous return branches, is very general, is similar to the Everett theory [13] and moreover does not impose constraints on the expression of $P(\alpha, \beta)$. However, it requires a large amount of measured data. A reduction of the input data, needed for the identification, can be obtained by making some assumptions, justified on physical basis for a given class of magnetic materials [14]. In particular, the assumption of the factorization $P(\alpha, \beta) = p(\alpha)p(-\beta)$, allows the determination of P by means of the procedure proposed by Kadar [3], [15].

Finally, the electromagnetic fields inside the lamination are modelled with a one dimensional diffusion equation, solved numerically by finite element and time stepping techniques. Taking the x-axis perpendicular to the lamination, the equation reads

$$\frac{1}{\sigma(x)}\frac{\partial^2 H}{\partial x^2} = \mu_0 \frac{\partial H}{\partial t} + \frac{\partial M}{\partial t}, 0 < x < d/2 \tag{8}$$

Notice that $H_s(t)$ and $B_a(t)$, introduced in (4) and (5), are given by

$$H_s(t) = H(x = d/2, t), B_a(t) = \frac{2}{d}\int_0^{d/2} (\mu_0 H(x, t) + M(x, t)) dx, \tag{9}$$

The relation between $M(x,t)$ and $H(x,t)$ is described by the DPM (6), (7).

Experimental evaluation and concluding remarks

Now, we discuss the validity of the combined lamination-dynamic Preisach model (LM-DPM) (6)-(7)-(8) with experimental results. For the validation we considered a classical 3.2% laminated SiFe alloy of thickness 0.35 mm for which the material parameters $P(\alpha,\beta)$, k_m, k_n and k_d are identified, cf. [2], [12]. Moreover, a simplified model is considered, which starts from the B_aH_h-relation, obtained from the rate-independent model corresponding with

$$\phi = \begin{cases} \frac{H_e(t)-\alpha}{k_n} & , H_e(t) > \alpha + k_n\phi \text{ and } \phi < +1 \\ \frac{H_e(t)-\beta}{k_n} & , H_e(t) < \beta + k_n\phi \text{ and } \phi > -1 \\ +1 & , H_e(t) \geq \alpha + k_n \\ -1 & , H_e(t) \leq \beta - k_n \end{cases} \tag{10}$$

and describing the quasi-static (hysteresis) behaviour of the material. To obtain the B_aH_s-loop, the hysteresis field H_h is corrected by the dynamic field $H_d = H_c + H_e$ at each time point t, in order to obtain the magnetic field at the surface of the lamination, viz $H_s = H_h + H_d$. The function $H_d(B_a, dB_a/dt)$ is identified experimentally, see [16]. Here, the electrical conductivity σ and the PDF in (8) is assumed to be x-independent.

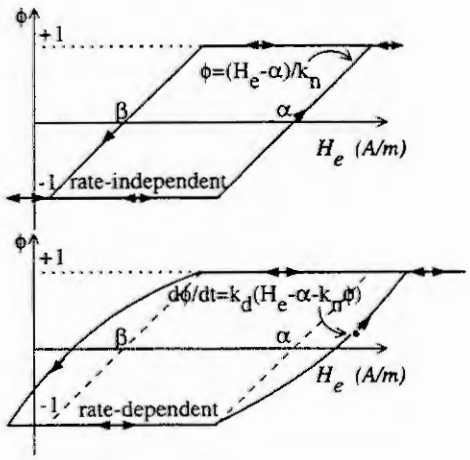


Figure 1: Preisach dipoles characteristic

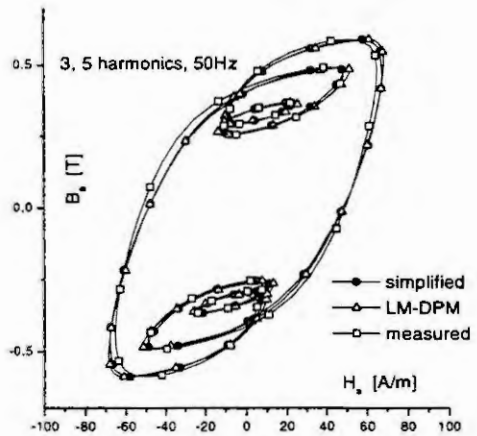


Figure 2: Comparison measured and calculated B_aH_s-loops with 3rd and 5th harmonics

As higher harmonics have practical relevance towards the analysis of electromagnetic devices, we consider first a B_a-excitation of 50Hz-excitation with 3rd and 5th harmonics (representing saturation effects) superponed, next a B_a-excitation of 50Hz-excitation with 17th and 19th harmonics (which are due to teeth effects). In Fig.2 we compare the B_aH_s-loops obtained by the simplified method, by the LM-DPM and by measurements for the case of the 3rd and 5th harmonics, while Fig.3 gives a similar comparison for the case of the 17th and 19th harmonics. The B_aH_s-loops obtained by the simplified method and by the LM-DPM are in good agreement with the experimental B_aH_s-loops.

Finally, we investigate the influence of a varying Si% throughout the thickness of SiFe laminations on its magnetic performance under unidirectional flux excitations. We take a %Si which varies linearly between the value 0.5 in the middle of the lamination to the value 4.0 at each of the surfaces. Indeed, eddy currents, mainly appearing at the surface of the lamination due to skin effects, may be reduced by increasing locally the resistivity (depending linearly on the %Si). As the material parameters are depending on x, the simplified model becomes inappropriate. Fig.4 shows the space variation of the rms values of the eddy currents for the following cases: a uniform 0.5 %Si, a uniform 4.0%Si and a linearly varying %Si between 0.5 and 4.0. It is observed that due to a varying Si%, the rms values change in a strongly nonlinear way.

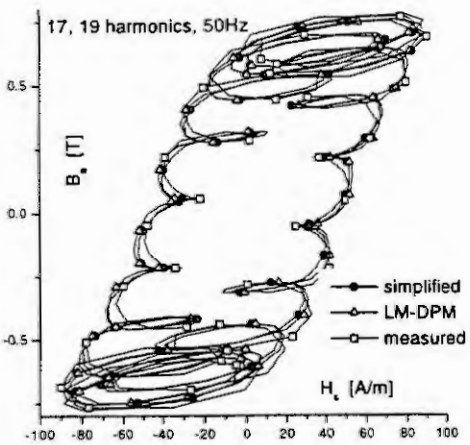


Figure 3: Comparison measured and calculated B_aH_s-loops with 17th and 19th harmonics

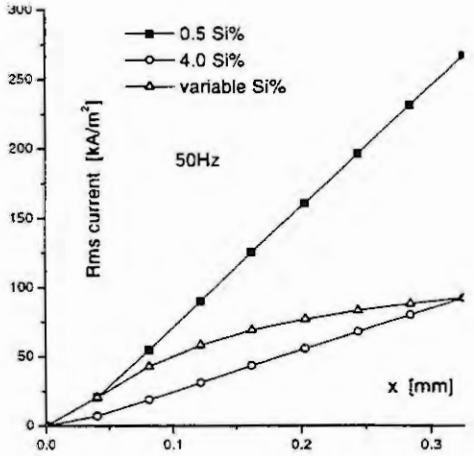


Figure 4: Space dependency of the eddy currents resulting in the classical losses

Acknowledgments

The first author is post-doctoral researcher of the Fund of scientific research-Flanders. (F.W.O.-Vlaanderen).

References

[1] Preisach, F., Uber die magnetishe Nachwirkung. Zeits. fur Phys., 94 (1935), 277-302.

[2] Mayergoyz I.D., Mathematical models of hysteresis Springer Verlag, Berlin, 1991

[3] Kadar, G., The Preisach function of ferromagnetic hysteresis. J. Appl. Phys., 61 (1987), 4013-4015.

[4] Della Torre, E., Effect of interaction on the magnetisation of single domain particles. IEEE Trans. Audio, 14 (1966), 86-93.

[5] Bertotti, G., General Properties of Power Losses in Soft Ferromagnetic Materials, IEEE Trans. Magn., 24 (1988), 621-630.

[6] Bertotti, G., Space-time correlation properties of the magnetisation process and eddy current losses: applications, I. Fine wall spacing. J. Appl. Phys., 55 (1984), 4339-4347.

[7] Dupré, L., Bertotti, G., Melkebeek, J., Dynamic Preisach model and energy dissipation in soft magnetic materials, IEEE Trans. Magn., 34 (1998), 1168-1170.

[8] Bertotti, G., Dynamic generalization of the scalar Preisach model of hysteresis. IEEE. Trans. Magn., 28 (1992), 2599-2601.

[9] Bertotti, G., Generalized Preisach model for the description of hysteresis and eddy current effects in metallic ferromagnetic materials. J. Appl. Phys., 69 (1991), 4608-4610.

[10] Del Vecchio, R., An efficient procedure for modeling complex hysteresis processes in ferromagnetic materials. IEEE Trans. Magn., 16 (1980), 809-811.

[11] Bertotti, G., Di Schino, G., Ferro Milone, A., Fiorillo, F., On the effect of grain size on magnetic losses of 3% SiFe. J. de Phys., 46 (1985), 385-388.

[12] Dupré, L., Bertotti, G., Basso, V., Fiorillo, F., Melkebeek, J., Generalisation of the dynamic Preisach model towards grain oriented Fe-Si alloys. Physica B: Physics of Condensed Matter (accepted).

[13] Everett, D., A general approach to hysteresis-part 4. An alternative formulation of the domain model. Trans. Faraday Soc., 51 (1955), 1551-1557.

[14] Basso, V. et al., Preisach model study of the connection between magnetic and microstructural properties of soft magnetic materials. IEEE Trans. Magn., 31 (1995), 4000-4005.

[15] Kadar, G., Della Torre, E., Determination of the bilinear Product Preisach function. J. Appl. Phys., 63 (1988), 3001-3003.

[16] Dupré, L., Van Keer, R., Melkebeek, J., Simplified models for the evaluation of the electromagnetic behaviour of SiFe alloys using the Preisach theory. J. of Eng. Science (submitted)

NUMERICAL LENGTH SCALE PROBLEMS IN MIXED EULERIAN-LAGRANGIAN MODELING

E. Helland, R. Occelli and L. Tadrist
IUSTI - CNRS (UMR 6595) - University of Provence - Technopôle de Château Gombert
5, rue Enrico Fermi - 13453 Marseille Cedex 13 - T: 33-4 91 10 69 36 - F: 33-4 91 10 69 69
E: evin@iusti.univ-mrs.fr

Abstract. Simulations with two-way coupling are performed for two dimensional gas-solid flow. The motion of particles is treated by a Lagrangian approach and particles are assumed to interact through binary, instantaneous, non-frontal, inelastic collisions with friction. The model for the interstitial gas phase is based on the Navier-Stokes equations for two-phase flow. Several porosity functions exist in the literature relating the drag force for a particle in a cloud to the drag force on an isolated particle. Thus it is important to have a correct definition of the porosity when modeling such systems. The fluid control volume must be little enough to minimize discretization errors, however, at the same time it must be large enough to have a representative elementary volume for the porosity estimation in order to reduce physical modeling errors. The influence of the grid size, thus the porosity definition, has been studied showing significant differences in the two-phase flow structure.

Introduction

The hydrodynamics of gas-particle fluidized beds have attracted great interest during recent decades due to their widespread applications in industries. The numerical simulation plays an important role in the prediction of the flow behaviour in fluidized beds. Both Eulerian and Lagrangian approaches can be used to describe the system. At present the Eulerian description for both the solid and gas phase is most developed, but interest in the mixed Eulerian-Lagrangian approach (Tsuji [1], Hoomans *et al.* [2]) is growing as computional capacity increases. The presence of each phase is described by a volume fraction, varying from zero to one. The gas fraction in such gas-particle flows is often called the porosity.

In order to better understand the accuracy of numerical simulations of fluid dynamics, the fundamental sources of inaccuracies must be identified. Error sources can be grouped into four broad categories (Oberkampf and Blottner [3]): 1) physical modeling errors, 2) discretization and solution errors, 3) programming errors and 4) computer round-off errors. Spatial and temporel resolution is an important source to errors in our modeling, however, an accurate definition of the porosity in two-phase flows is crucial in order to reduce physical modeling errors. The fluid control volume must be little enough to minimize discretization errors, on the other hand, it must be large enough to have a *representative elementary volume* (REV) for the porosity estimation to permit the meaningful statistical average required in the continuum concept. For a matrix made of uniformly sized particles that are regularly arranged, this REV includes only a few particles. The influence of the grid size, thus the porosity definition, has been studied showing significant differences in the two-phase flow structure. These results point out significant problems we meet when dealing with multi-phase flows having different caracteristic length and time scales.

Physical model

The Eulerian/Lagrangian method computes the Navier Stokes equation for the gas phase and the motion of invidual particles by the Newtonian equations of motion. For the gas phase, we write the equations of conservation of mass and momentum :

$$\frac{\partial(\varepsilon\rho_g)}{\partial t} + \nabla\cdot(\varepsilon\rho_g\bar{u}) = 0 \tag{1}$$

$$\frac{\partial(\varepsilon\rho_g\bar{u})}{\partial t} + \nabla\cdot(\varepsilon\rho_g\bar{u}\bar{u}) = -\varepsilon\nabla\cdot(p\bar{I}) + \mu_g(\nabla\cdot\varepsilon\bar{\tau}_g) + \varepsilon\rho_g\bar{g} - \sum_{i=1}^{Np(V)}\bar{f}_{drag,i} \tag{2}$$

In discrete particle models for each individual particle an equation of motion is solved during the free flight phase :

$$m_i\frac{d\bar{v}_i}{dt} = m_i\bar{g} + \bar{F}_{drag} \tag{3}$$

where m_i, v_i represent respectively the mass and velocity of the i^{th} particle and the right hand side the sum of the forces acting on the i^{th} particle; the first term is due to gravity and the second due to drag between the gas and

particle phase. The pressure gradient force, the buoyancy force and the unsteady forces have been neglected due to the high solid to gas density ratio. The slip/rotation or Magnus lift force and the slip/shear or Saffman lift force have also been neglected due to the small particle diameter. The drag force is quantified through the equation :

$$\vec{F}_{drag} = \frac{C_d}{8}\pi d_p^2 \rho_g |\vec{u} - \vec{v}_i|(\vec{u} - \vec{v}_i)\varepsilon^2 f(\varepsilon) \tag{4}$$

The drag coefficient C_d on a single sphere is given by Schiller and Naumann [4] :

$$C_d = \begin{cases} 24\left(1 + 0.15\,\mathrm{Re}_p^{0.687}\right)/\mathrm{Re}_p & \mathrm{Re}_p < 1000 \\ 0.44 & \mathrm{Re}_p \geq 1000 \end{cases} \tag{5}$$

A number of empirical relationships of the drag coefficient have been proposed in the litterature. However, there is little information available on the drag of particles in particle clusters. Analytical models are difficult, because the surface of every particle must be taken into account. In regions with few particles distributed non-uniformly the descent of a given particle can create a velocity field throughout the fluid which tends to decrease the drag of all particles near it due to bypassing of return flow. On the other hand, if the particles are more or less uniformly distributed through the fluid, the restriction of the flow spaces between the particles in denser zones results in steeper velocity gradients of the gas phase, thus greater shearing stresses and an increase in resistance of the gas flow. The shape of the flow conduits between the particles, and hence the flow pattern is a function of the ratio of the particle diameter d_p to the distance l_p between the particles. For a uniformely dispersed suspension, d_p / l_p is a function of the porosity ε only. This influence has often been taken into account in numerical studies of gas-particle flows by using the experimental results on sedimentation and fluidization of Richardson and Zaki [5]:

$$f(\varepsilon) = \varepsilon^{-n} \tag{6}$$

with

$$n = \frac{\log\left(\dfrac{\mathrm{Re}_{mf}}{\mathrm{Re}_t}\right)}{\log \varepsilon_{mf}} \qquad (n \in \langle 2,10 \rangle) \tag{7}$$

The equation (3) can be rewritten as :

$$\frac{d\vec{v}_i}{dt} = \vec{g} + \frac{1}{\tau_{ig}}(\vec{u} - \vec{v}_i) \tag{8}$$

with

$$\tau_{ig} = \frac{\rho_p d_p^2}{18\mu_g}\varepsilon^{n-1} \tag{9}$$

The collision model proposed by Walton [6] is used to compute the dynamics of instantaneous inelastic non-frontal collisions with friction based on three constant coefficients. The first coefficient e characterizes the incomplete restitution of the normal component of the relative velocity at the point of contact. The second μ arises in collisions involving sliding and has been assumed to be resisted by Coulomb friction. The third β arises in collisions that return a fraction of the energy stored in the elastic deformation of both surfaces to the component of the contact velocity tangent to the spheres. Foerster *et al.* [7] showed experimentally that the model captures the behaviour of the impact between glass spheres over a wide range of incident angles. The collision parameters used in the present study is taken from these experimental data.

Calculation of porosity

The porosity in a computational cell is the ratio of the gas volume to the volume of the fluid control volume ΔV. If V_i is the volume of particle i inside a computational cell, then the porosity in this cell is

$$\varepsilon = 1 - \sum_{i=1}^{N_c} V_i \Big/ \Delta V. \tag{10}$$

A pseudo-three-dimensional concept is employed since two-dimensional simulations are performed in this study. We suppose that the simulations is performed in a three-dimensional bed with its thickness equal to the diameter of the spherical particles.

Solution technique

Equations (1) and (2) are solved by a finite volume method. The well known SIMPLE scheme is used as iterative solution procedure (Patankar [8]). A staggered grid is used with velocity components stored at the

control volume surfaces and the scalar variables in the center. The integration of the conservation equations is performed using the power law/Quick scheme in space and an implicit scheme in time. A semi-implicit intergration method is used to solve the equation of particle motion (8). In the model two time scales are distinguished which means that two different time steps are used. A *fluid time step* Δt is used to solve the gas phase motion, and within this one several *particle time steps* Δt_p for the hard-sphere particle simulations are performed. The fluid force time step Δt is chosen to be smaller than the particle relaxation time at maximum packing concentration (10^{-3} s). In order to save CPU time several *particle control volumes* (PCV) within the *fluid control volume* (FCV) is used. The fluid control volume is taken large enough to have a representative elementary volume, but little enough to have small changes in flow properties. The surface of the PCVs is chosen to order the particle surface. A minimum particle time step is defined such that any particle having the actual maximum gas velocity can only pass 20 % of the minimum length of the PCV. This calculated time step is used if it is less than the maximum particle time step Δt_p defined by user. This method seems to be efficient as no or an insignificant fraction of particle overlapping is observed even in zones of high solid fraction.

The gas motion is calculated implicitely using the equation (1) and (2) at time step $t+\Delta t$ with the explicit drag term estimated at time step t. Then the particle positions are calculated for the next *particle time step* $t+\Delta t_p$ using equation (8) and then a check for overlapping between particles entirely within the PCVs is performed and then between particles in two adjacent PCVs. Once an overlapping between two particles has been detected the particles are placed in their previous positions. Then the collision dynamics is calculated inducing a change of the translation and rotation velocities and the position of the two particles. The particle's corresponding gas velocity (at time step $t+\Delta t$) and porosity (at time step t) is calculated by linear interpolation within the FCV. After finding the new velocities and positions of all particles at the time step $t+\Delta t$, the porosity is estimated and the drag forces on every particle summarized within each FCV, before continuing a new computational cycle.

Results and discussion

The air inlets are modelled as one-dimensional uniform flow 2 mm in front of the air distributor in order to have an air flow at the distributor level with small amplitude variations due to the motion of particles above the distributor. The particles leaving at the top are simultaneously introduced at the bottom of the riser at random positions with zero velocities. A no-slip condition is used for the gas phase at the walls and the particles are allowed for frontal collisions at the wall. The simulations were run for 2 seconds real time. The parameters settings are the following: u_y = 0.05 m/s, H = 1.2 cm, D = 0.12 cm, $\Delta t = 10^{-4}$ s, $\Delta t_p = 10^{-5}$ s, $\mu_g = 1.8*10^{-5}$ kg/m/s, ρ_g = 1.2 kg/m^3, ρ_p = 2000 kg/m^3, d_p = 50 µm, n = 4.7, e = 0.97, e_{wall} = 0.85, γ = 0.1, β = 0.5 et N_p = 300. The fluid Reynolds number is close to 5 and the terminal particle Reynolds number is 0.5.

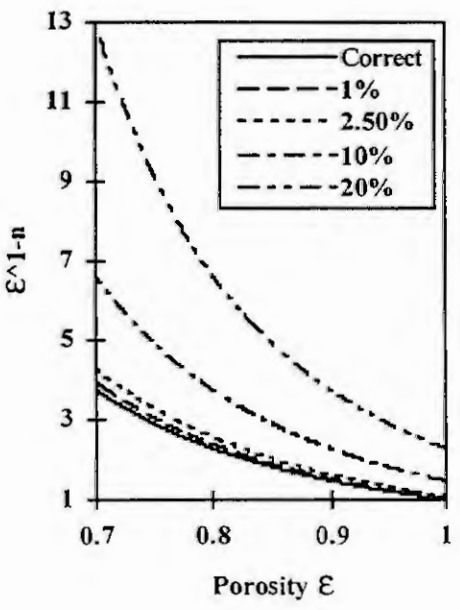


Figure 1. *Influence of a x % under-estimation of the porosity on the drag force (eq. 8).*

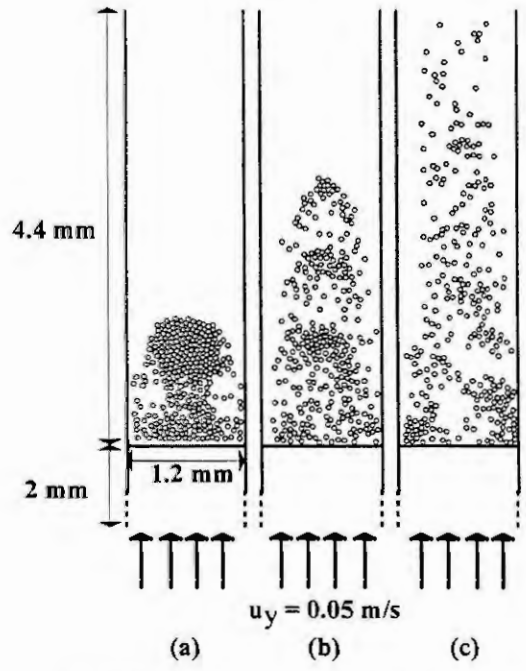


Figure 2. *Influence of the FCV size on the flow behaviour in a fluidized bed.*

We showed previously that the particle relaxation time decreases exponentially as the porosity decreases indicating that the drag force influences the gas-particle flow radically stronger as solids concentration increases. If the fluid control volume is much greater than the particle volume, we have a representative elementary volume for the porosity estimation, however, we loose accurancy due to discretization errors and we have small local variations of fluid properties. If a single particle occupy a significant volume of the fluid control volume, we under-estimate the porosity herein, thus an over-estimation of its drag. From figure 1 it can be observed that an under-estimation up to 2.5% of the porosity has insignificant effects on the drag force for particles in assemblages. The influence of the size of the fluid control volume is shown in figure 2 and 3: the ratio of the single particle volume to fluid control volume: (a) 1%, (b) 2.5% and (c) 20%. Small variations of gas-particle properties are observed for large *FCVs* (a) and the particles rest nearly immobile in the bottom of the bed as gravity dominates the drag. Smaller *FCVs* (b) give a more realistic picture of the two-phase motion for bubbling fluidized beds where gas-particle variables undergo smooth large scale oscillations. These fluctuations are due to the influence of the particle assemblage effect on the local drag as small perturbations of the porosity give rise to important drag variations as discussed previously (eq. 8-9). When reducing the *FCV* further (c), we observe that the particles are suspended in the air as drag equals the gravity due to an over-estimation of the drag. This is quite wrong as the terminal velocity for isolated particles is close to 0.15 m/s. The reduction of the *FCV* induces an important jump in porosity as one particle enters or leaves it, thus sharp fluctuations of the gas-particle variables are observed.

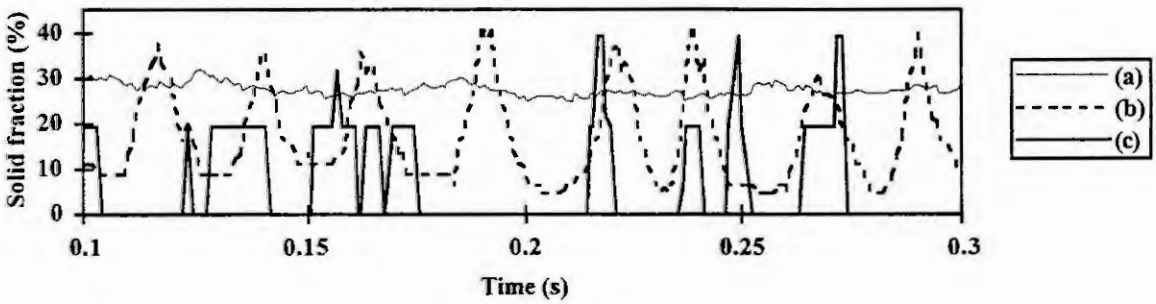


Figure 3. *Influence of the FCV size on the solid fraction evolution at a fixed point in the riser.*

Conclusions

The influence of the particle volume to fluid control volume ratio, thus the porosity definition, has been studied showing significant differences in the two-phase flow structure. Intermediate sizes of fluid control volume must be used in order to minimize both discetization and physical modeling errors. These results point out significant problems we meet when dealing with multi-phase flows having different caracteristic length and time scales.

Acknowledgments

We are grateful to Agence de l'Environnement et de la Maîtrise de l'Energie (ADEME) for financial assistance.

References

1. Tsuji, Y., KONA, 11 (1993), pp. 57-68
2. Hoomans, B.P.B., Kuipers, J.A.M., Briels, W.J., Van Swaaij, W.P.M., Chem. Eng. Sci., 51 (1996), pp. 99-118
3. Oberkampf, W. L., Blottner, F. G., AIAA Journal, 36, No. 5 (1998), pp. 687
4. Schiller, L., Naumann, A., Z., Ver. Deut. Ing., 77 (1935), pp. 318
5. Richardson, J.F., Zaki, W.N., Trans. Inst. Chem. Eng., 32 (1954), pp. 35
6. Walton, O.R., Quarterly Report Jan-Mar 1988, UCID-20297-88-1, Lawrence Livermore National Laboratory (1988)
7. Foerster, S. Louge, M., Chang, H., Allia, K., Phys. Fluids, 6 (1994), pp. 1108
8. Patankar, S. V., Numerical Heat Transfer and Fluid Flow, Hemisphere Publishing Corporation (1980)

LYAPUNOV FUNCTION FOR AN INDUCTION GENERATOR – INFINITE BUS POWER SYSTEM WITH TRANSMISSION LOSSES

Josiah L. Munda and Hayao Miyagi
Information Engineering Department, University of the Ryukyus
1 Senbaru, Nishihara, Okinawa 903-0213 Japan
E-mail: munda@sys.ie.u-ryukyu.ac.jp

Abstract. This paper deals with the dynamic and transient stability analysis of an induction generator - infinite bus power system. A Lyapunov function is developed from the nonlinear differential equations of the power system, by expressing the electromotive force (emf) of the generator rotor as a function of slip. The mathematical models are derived by assuming that the mechanical power input to the generator is from a wind turbine. Self-excitation of the induction generator is from a terminal capacitor, the value of which is included in the mathematical models.

1. Introduction.

The purpose of a power system is to deliver the power required by the consumers in real time, on demand, within acceptable voltage and frequency limits, and in a reliable and economic manner. In remote areas and islands power networks are usually not connected to the grid, and electrical energy is generally generated using diesel fuel. The cost of producing a unit of electrical energy that way is found to be very high, thus the need for using renewable sources of energy e. g wind, tidal, natural gases, geothermal etc. Another factor is the increasing global concern for protecting the environment. Wind power is attracting wide attention as a clean and environmentally- friendly renewable energy.

The property of an induction machine of being able to be used to produce electrical energy at a wide range of speeds finds particular application in the generation of electrical energy at constant frequency from wind energy, since wind has a rather unsteady speed characteristic. Different induction generator schemes have been employed, depending on the application areas. For autonomous systems the self-excited induction generator is the most commonly used [4]. One means of achieving self-excitation is through the connection of a terminal capacitor.

Studies concerned with the dynamics of electrical systems involve the solution of sets of nonlinear differential equations for the systems' fault-on and post-fault states using numerical methods through digital simulations. The direct method of Lyapunov gives a direct solution to the above problem so as to arrive at the critical clearing time for the faults on the system. A Lyapunov function is constructed from the system's state variables about the post-fault stable equilibrium point. All research work so far on power system stability using the direct method has been dealing only with systems supplied from synchronous generators. Use of induction generators is increasing, hence the need for their proper analysis especially in large systems.

A power system is transiently stable for a particular steady-state operating condition and for a particular large disturbance if, following that disturbance, it reaches an acceptable steady-state operating condition [3]. The main factors on which transient stability depends are; load-generation balance, nature and location of disturbance, and network configuration before fault occurrence and after its subsequent clearance. The main problem is to come up with appropriate models for the system under different operating conditions, for use in simulation studies.

This paper proposes an energy function for a power system supplied by an induction generator, driven by a wind turbine. The mathematical modelling of the system takes into account the dependence of input power on wind speed and slip, and the changes in self-excitation. Use is then made of a relationship between the rotor angle and slip, and the general mechanical equation of motion of the induction generator.

2. Modelling

The mathematical models of the system are derived based on the assumptions that the network is initially in sinusoidal steady state, the phase angle of the voltage coincides with the rotor angle, and the mechanical torque is a function of wind speed and slip.

Neglecting the dc component in the stator transient currents, the dynamic models of the induction generator are derived as [3];

$$2H\frac{ds}{dt} = T_e - T_m \tag{1}$$

$$\frac{d\dot{E}}{dt} = -\frac{1}{T_o'}\left[\dot{E} + j(x-x')\dot{I}\right] - j\omega_0 s\dot{E} \tag{2}$$

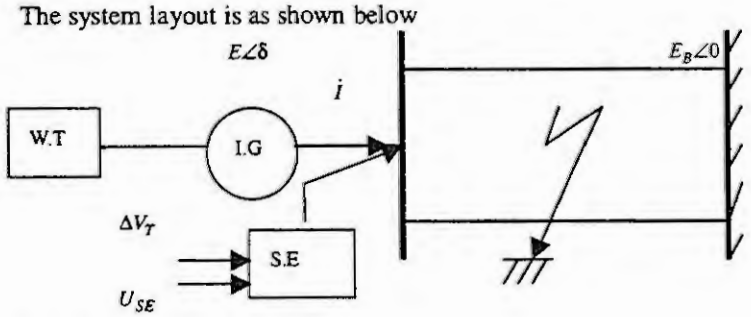
where $\dot{E}$ = complex emf behind transient reactance, corresponding to rotor flux linkages [p. u]; $\dot{I}$ = induction generator current [p. u]; H = inertia constant of the generator [s]; T_e = electromagnetic torque developed[p. u]; T_m = mechanical torque input[p. u]; T_0' = transient open-circuit time constant [s]; x' = generator transient reactance [p. u]; x= sum of stator and magnetizing reactance [p. u]; t= time [s]; and

$$s = (\omega_0 - \omega)/\omega_0 \text{ , slip [p.u]} \tag{3}$$

where ω_0 and ω are synchronous speed and rotor angular speed respectively [rpm]

In this paper we shall assume that the terminal voltage of the induction generator is held constant through some control means, acting on the self-excitation capacitor. It has been shown that the self-excited induction generator can be used to generate electrical power at constant voltage and frequency with varying wind speed and load conditions [2,4].

The system layout is as shown below



Here,
I.G= induction generator,
W.T= wind turbine
S.E= self-excitation system
ΔV_T = change in terminal voltage
U_{SE} = excitation control input

Fig.1 System layout

For easier analysis the system is transformed into an equivalent π-model as given in fig.2 below.

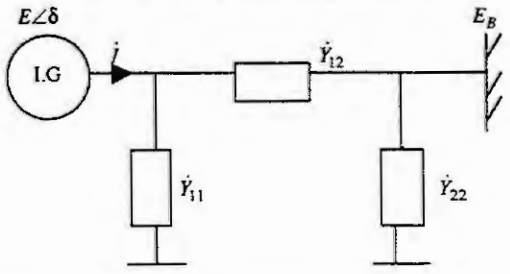


Fig.2 Equivalent π-network.

The value of the generator current is obtained as,

$$\dot{I} = \dot{E}\dot{Y}_{11} + (\dot{E} - E_B)\dot{Y}_{12} \tag{4}$$

where $\dot{Y}_{ij} = G_{ij} + jB_{ij}$ (5)

and $\dot{Y}_{ij}, B_{ij}, G_{ij}$ are admittance, conductance and, susceptance of the respective elements [p. u].

Through Y/Δ transformation, we can obtain the conductance and susceptance values in the form;

$$G_{11}=\frac{B_{SE}\left[G_g B_{SE}+a_3\right]}{a_1{}^2+(a_2+B_{SE})^2};G_{12}=\frac{a_5+a_4(a_2+B_{SE})}{a_1{}^2+(a_2+B_{SE})^2};B_{11}=\frac{B_{SE}\left[B_g B_{SE}+a_6\right]}{a_1{}^2+(a_2+B_{SE})^2};B_{12}=\frac{a_8-a_7(a_2+B_{SE})}{a_1{}^2+(a_2+B_{SE})^2} \quad (6)$$

Here, B_{SE}, B_g, G_g are susceptance of the self-exciter, susceptance and conductance of the generator respectively;

$$a_1=G_g+G_l;a_2=B_g+B_l;a_3=G_g B_l-B_g G_l;a_4=G_g B_l+B_g G_l;$$

$$a_5=a_1\left(G_g G_l-B_g B_l\right)a_6=B_g{}^2+G_g{}^2+B_g B_l+G_g G_l;a_7=a_5/a_1;a_8=a_1 a_4$$

Let us express the generator emf as,

$$\dot{E}=E_d+jE_q=E\exp(j\delta) \quad (7)$$

The generator current can then be simplified to the form;

$$\dot{I}=\left(E_d(G_{11}+G_{12})-E_B G_{12}-E_q(B_{11}+B_{12})\right)+j\left(E_d(B_{11}+B_{12})-E_B B_{12}+E_q(G_{11}+G_{12})\right) \quad (8)$$

The output power from the generator is calculated as;

$$P_e=\mathrm{Re}\left[\dot{E}'\dot{I}^*\right] \quad (9)$$

where $\dot{I}^*$ is the complex conjugate of system current [p. u]

Substituting the values of current and voltage;

$$P_e=E^2(G_{11}+G_{12})-EE_B(G_{12}\cos\delta+B_{12}\sin\delta) \quad (10)$$

The electromagnetic torque of the induction generator is equal to;

$$T_e=P_e/\omega \quad (11)$$

Let us express the mechanical power input in the form;

$$P_m=P_w-D_g\omega^2 \quad (12)$$

where D_g, P_w are the damping coefficient of the system, and the aerodynamic power generated by the rotor respectively.

$$P_w=\frac{1}{2}k_p\rho AR\omega V_w{}^2=k\omega V_w{}^2 \quad (13)$$

where k_p, ρ, A, R, V_w are a constant of the turbine, the air density, the area swept by turbine rotor, the rotor radius, and the wind speed respectively.

The mechanical torque therefore becomes,

$$T_m=\frac{P_m}{\omega}=kV_w{}^2-D_g\omega \quad (14)$$

The mathematical models for the system can now be written in terms of the d-axis component of emf as;

$$2H\frac{ds}{dt}=\frac{E_d{}^2(G_{11}+G_{12})\sec^2\delta-E_d E_B\left[G_{12}+B_{12}\tan\delta\right]}{\omega_o(1-s)}-\left[kV_w{}^2-D_g\omega_o(1-s)\right] \quad (15)$$

$$\frac{dE_d}{dt} = -\frac{1}{T_o'}\left[E_d - (x-x')\left[E_d\,(G_{11}+G_{12})\tan\delta + E_d\,(B_{11}+B_{12}) - E_B B_{12}\right]\right] + \omega_o s E_d \tan\delta \qquad (16)$$

Next, to try to maintain a constant terminal voltage, we introduce the equation for the variation of the self-excitation susceptance;

$$\frac{d\Delta B_{SE}}{dt} = -\frac{1}{T_{SE}}\left[\Delta B_{SE} + K_{SE}\Delta V_T - U_{SE}\right] \qquad (17)$$

where T_{SE}, K_{SE} are time constant and control gain of the self-exciter respectively.

$\Delta V_T = V_{SE} - V_{SE0}$, and $U_{SE} = f(\delta, s)$

Let us introduce an expression between rotor angle and slip of the induction machine in the form [1];

$$\frac{d\delta}{dt} = \omega_0 (s_0 - s) \qquad (18)$$

where s_0= rated (steady state) slip

The dynamics of the power system can then be studied by obtaining time solutions to the resulting system of nonlinear differentials equations, (15)-(18) above.

3. Lyapunov Function

To derive the Lyapunov function, we introduce state variables (the dot notation now used for time-derivatives);

$$z_1 = \delta - \delta_0;\ z_2 = \dot{z}_1;\ z_3 = E_d - E_{d0};\ z_4 = B_{SE} - B_{SE0}$$

where $\delta_0, E_{d0}, B_{SE0}$ are steady state values.

Writing that, $V_T = V_d \sec\theta_v$, and assuming that the angle is almost constant, we may obtain an expression for the change in voltage as,

$$\Delta V_T = b_1 (E_d - E_{d0}) + b_2\left[E_d \tan\delta - E_{d0}\tan\delta_0\right] \qquad (19)$$

where $b_1 = \left[1 - r_g\,(G_{11}+G_{12}) - x'(B_{11}+B_{12})\right]\sec\theta_v;\ b_2 = \left[G_{11}+G_{12}-B_{11}-B_{12}\right]\sec\theta_v$

The mathematical models for the system can then be expressed in variable-state form as;

$$\dot{z}_1 = z_2 \qquad (20)$$

$$2H\dot{z}_2 = \omega_0 k V_w{}^2 - \omega_0{}^2 (1-s_0) D_g - \left[(z_3+E_{d0})^2 (G_{11}+G_{12})\sec^2(z_1+\delta_0) - E_B (z_3+E_{d0})\left[G_{12} + B_{12}\tan(z_1+\delta_0)\right]\right]/(1-s_0) \qquad (22)$$

$$T_0'\dot{z}_3 = -\left[1-(x-x')(B_{11}+B_{12})\right](z_3+E_{d0}) + \left[(x-x')(G_{11}+G_{12}) + \omega_o s_0 T_0'\right](z_3+E_{d0})\tan(z_1+\delta_0) - E_B B_{12} \qquad (23)$$

$$T_{SE}\dot{z}_4 = -z_4 - K_{SE} b_1 z_3 + f_u(z_1; z_2) - K_{SE} b_2\left[(z_3+E_{d0})\tan(z_1+\delta_0) - E_{d0}\tan\delta_0\right] \qquad (24)$$

The energy (Lyapunov) function for the system can be obtained (the details of which are omitted here) in terms of the physical variables from (20) – (24) as,

$$V = H\omega_0^2(1-s_0)(s_0-s)^2 - \omega_0(1-s_0)\left[kV_w^2 - \omega_0 D_g(1-s_0)\right](\delta-\delta_0) + E_d^2(G_{11}+G_{12})[\tan\delta - \tan\delta_0]$$
$$- E_B E_d\left[G_{12}(\delta-\delta_0) + B_{12}\left[\ln|\cos\delta| - \ln|\cos\delta_0|\right] + \frac{1}{2}\left[1-(x-x')(B_{11}+B_{12})\right]\left(|E_d|^2 - |E_{d0}|^2\right)\right. \quad (25)$$
$$\left. + \left[\frac{a_2a_4 + a_5 - G_g\left(a_1^2 + a_2^2\right)}{a_1}\right]\left[\tan^{-1}\left(\frac{B_{SE}+a_2}{a_1}\right) - \tan^{-1}\left(\frac{B_{SE0}+a_2}{a_1}\right)\right] + G_g(B_{SE}-B_{SE0})\right]$$

The time derivative of the obtained energy function can be shown to be negative-definite around the origin.

4. Simulation

Typical induction generator data is used to run tests to compare the rotor angles obtained from equations (7) and (18), through numerical integration of the system's nonlinear differential equations. The results are given by the curves of figure 3.

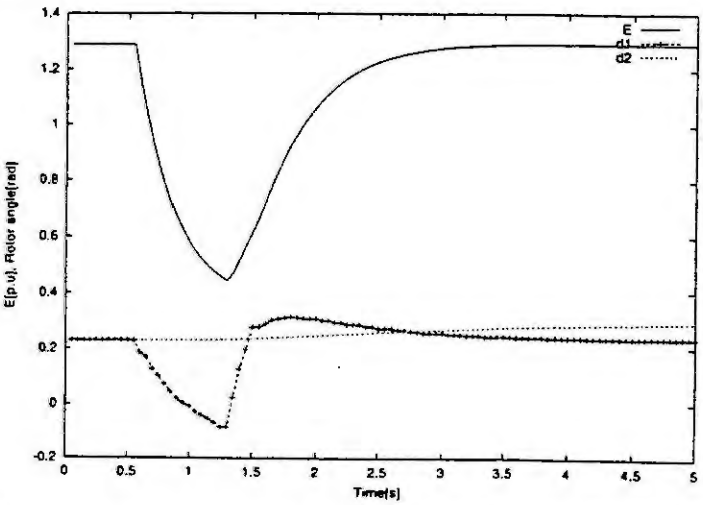


Fig 3. Variation of generator voltage and rotor angle with time following fault occurrence.

5. Conclusions

Mathematical models for a power system in which a wind-driven induction generator supplies an infinite bus have been derived. A Lyapunov function for the system is then developed from the obtained models. These results can be extended to the analysis of stability in multimachine power systems. This paper shows that the induction machine can also play an important role in stability analysis of power systems. Future work will deal with detailed modelling of control elements in the system for a more accurate analysis.

Acknowledgements. The authors gratefully acknowledge the financial assistance from Okiden Sekkei Co., Incorporation, Okinawa.

References

1. Munda, J. L., Asato, S. and Miyagi, H., Lyapunov Functions for a Synchronous Generator – Induction Motor Power System. In: Proc. 3rd IASTED International Conference, Power and Energy Systems,Las Vegas, 1999, Anaheim, 1999, 438-442.
2. Salama, M. H. and Holmes, P. G, Transient and steady-state load performance of a stand-alone self- excited induction generator. In: IEE Proc.- Electr. Power Appl., Vol. 143, 1 (1996), 50-58.
3. Pavella, M. and Murthy, P. G, Transient Stability of Power Systems: Theory and Practice. Wiley, England, 1994.
4. Uctug, M. Y., Eskandarzabeh, I.and Ince, H., Modelling and output power optimization of a wind turbine driven double output induction generator. In: IEE Proc.- Electr. Power Appl., Vol. 141, 2 (1994), 33-38.

Modeling of the works water section of a power plant group

Mathias Meusburger[1], Kurt Schlacher[1] and Alfons Sillaber[2]
[1] Johannes Kepler University of Linz
Altenbergerstraße 69, A-4040 Linz, AUSTRIA
e-mail: mathias@regpro.mechatronik.uni-linz.ac.at
[2] Innsbrucker Kommunalbetriebe AG
Salurner Straße 7, A-6000 Innsbruck, AUSTRIA

Abstract. In this paper, we present the modeling of the work water section of a power plant group. The mathematical model is based on the Saint-Venant equations and the nonlinear partial differential equations for the one-dimensional pipe flow. The model has to deal with a special coupling between an open channel and a pipe flow. With appropriate assumptions the model can be reduced significantly, so that the final model consists of coupling of nonlinear ordinary and nonlinear partial differential equations. A comparison of measurement results with simulation results shows the good practical exactness of the proposed model.

Introduction

In this contribution the mathematical modeling of the works water section of a power plant group is presented. The problem definition is due to the electric supply company of the town Innsbruck. The company is implementing a remote control systems for their power plants and for the design of such a control system a mathematical model, which describes the dynamical behavior of the plant, is necessary. The object of investigation consists of a supply tunnel with a slope of 0.1% and a length of about 7 km, the surge chambers Sill and Ruetz and a junction canal with a length of about 300 m, which connects the two surge chambers (see figure 1). The water is taken from the river Sill and flows through the supply canal to the surge chamber Sill, from which one part of the water directly flows through the pressure pipe to the turbine of the power-station Sill and the other one flows through a junction canal to the surge chamber of the power-station Ruetz. Near the flowing–out end of the supply canal a sluice gate is situated in order to dam the water (see figure 1). The electric supply company makes use of the supply canal to store water during slack periods in order to use this stored water to cover load peaks. Now, the flow in the canal should be modeled in order to examine the behavior of the plant in cases of large changes in the flow rate and to be able to develop a controller for the sluice gate.

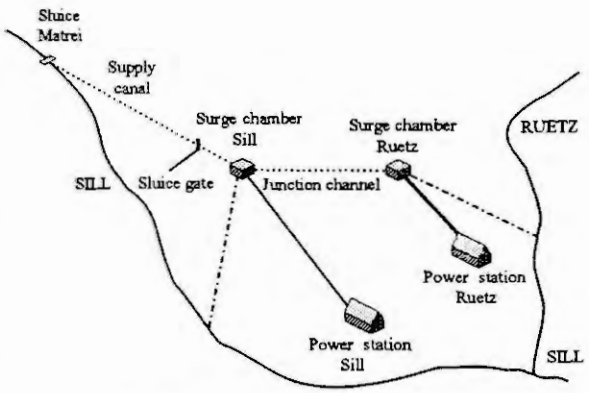


Figure 1: Schematic diagram of the plant.

Mathematical model

The mathematical model of the considered part of the plant is based on the well known Saint-Venant equations [5], [6], the *continuity equation*

$$A_t + Q_x = 0 \tag{1}$$

with the cross section A, the flow Q and the *momentum equation*

$$Q_t + 2\frac{Q}{A}Q_x + \left(gA - \left(\frac{Q}{A}\right)^2 A_y\right) y_x + gA(S - S_0) = 0 \tag{2}$$

with the depth y measured normally to the bottom, the gravity constant g, the canal slope S_0 and the friction S due to the Manning formula

$$S = \frac{n^2 Q\,|Q|}{A^2\,(A/P)^{4/3}}$$

with the flow Q, the Manning roughness factor n, the area A and the wetted perimeter P (e.g. [4], [5], [2]). The indices x, y and t denote the partial derivatives with respect to x, y and t, respectively.

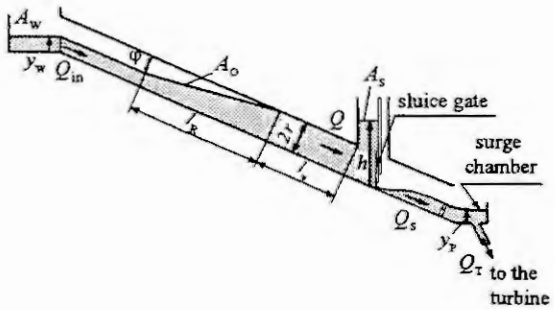


Figure 2: Flow in the canal.

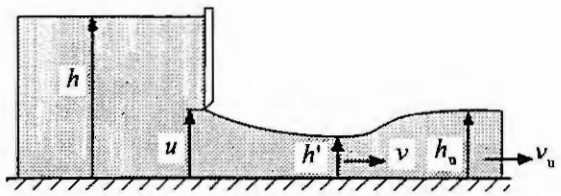


Figure 3: Flow under the sluice gate.

Depending on the water level before the sluice gate, the first part of the supply canal upstream can be completely filled with water and must therefore be treated as a pipe flow area. The second part has a free surface flow and the point, where there is this passing from the open channel to the pipe flow, changes. Since there is a big difference between the wave propagation velocity of the pipe flow area and the surface wave propagation velocity, this problem is inherently stiff. In order to avoid the well known difficulties of stiff systems, we propose an approach which is based on the following assumptions [3]:

1. The open channel flow area is considered as a huge damming area, where the surface is approximately plane.
2. The pipe flow is incompressible and the pipe is stiff, hence $A_x = A_t = Q_x = 0$.

Measurement results at the plant show that these assumptions are met in the plant of investigation. With these assumptions the mathematical model eq. (1) and eq. (2) simplifies to a system of ordinary differential equations of the form

$$\frac{\mathrm{d}Q}{\mathrm{d}t} = -gA\left(\frac{h-2r}{l_v} + S - S_0\right) \qquad \frac{\mathrm{d}l_v}{\mathrm{d}t} = \frac{Q_{in}-Q}{A_O \sin(\varphi)}$$

$$\frac{\mathrm{d}h}{\mathrm{d}t} = \frac{Q-Q_S}{A_S} \qquad \frac{\mathrm{d}y_w}{\mathrm{d}t} = \begin{cases} 0 & L > l_v + l_R \\ \dfrac{Q_{in}-Q}{A_W + A_O} & L < l_v + l_R \end{cases}$$

with the surface area A_O

$$A_O = \frac{1}{\sin(\varphi)}\left((r-y_w)\sqrt{y_w\,(2r-y_w)} + r^2 \arcsin\left(1-\frac{y_w}{r}\right) + \frac{r^2\pi}{2}\right),$$

the radius of the canal r, the depth in the chamber before the sluice gate h, the area of the chamber before the sluice gate A_S, the water level at the sluice Matrei y_w, the area of the sluice A_W, the length between the sluice gate and the point of change from the open channel to the pipe flow l_v, the length of the canal L, the estimated length of the backwater l_R, the flow under the sluice gate Q_S, the constant inflow Q_{in} and the angle φ as shown in figure 2. The schematic diagram of the simplified model of the upstream channel is depicted in detail in figure 2. The second part of the supply canal from the sluice gate downstream to the surge chamber Sill is always an open channel flow area. This part is modeled by

means of a spatial discretization of the corresponding partial differential equations (1) and (2) and we end up with a system of ordinary differential equations

$$\begin{aligned}\frac{\mathrm{d}Q_n}{\mathrm{d}t} &= -\left(gA_n - \left(\frac{Q_n}{A_n}\right)^2 A_{y,n}\right)\frac{\Delta y_n}{\Delta x_n} - 2\frac{Q_n}{A_n}\frac{\Delta Q_n}{\Delta x_n} - gA_n\left(S_n - S_0\right)\\ \frac{\mathrm{d}y_n}{\mathrm{d}t} &= -\frac{1}{A_{y,n}}\frac{\Delta Q_n}{\Delta x_n} \qquad \text{for} \qquad n = 0,...,N\end{aligned}$$

with $N+1$ points and the Dirichlet boundary conditions $y(N,t) = y_P(t)$ with the water level in the surge chamber Sill y_P and $Q(0,t) = Q_S(t)$ with the flow under the sluice gate Q_S.

Since the width of the junction channel changes, the approximation of the partial differential equations (1) and (2) by a finite difference scheme has to take this variation into consideration. We end up with the following system of ordinary differential equations

$$\frac{\mathrm{d}Q_n}{\mathrm{d}t} = -2\frac{Q_n}{A_n}\frac{\Delta Q_n}{\Delta x_n} - \left(gA_n - \frac{Q_n^2}{A_n^2}T_n\right)\frac{\Delta y_n}{\Delta x_n} + \frac{Q_n^2}{A_n^2}y_n\frac{\Delta b_n}{\Delta x_n} - gA_n\left(S_n - S_0\right) \qquad n = 1...N$$

$$\frac{\mathrm{d}y_n}{\mathrm{d}t} = -\frac{1}{A_{y,n}}\frac{\Delta Q_n}{\Delta x_n} \qquad \text{for} \qquad n = 0,...,N.$$

The variation of the canal width $\Delta b_n/\Delta x_n$ is chosen in such a way that the steady state solution of the partial differential equations coincides with the solution of the approximated system at the discretization points. Furthermore, we have to use Neumann boundary conditions, due to the fact that the water level in the surge chamber Ruetz can be lower than the bottom of the channel.

The mathematical model of the sluice gate (see figure 3) has to take into consideration both, the subcritical and the supercritical flow conditions. Following the standard approach in literature [1], [4] our model of the sluice gate is based on the Bernoulli equation (3), the impulse-momentum equation (4)

$$h + \frac{Q_S^2}{2g\left(hb_1\right)^2} = h' + \frac{Q_S^2}{2g\left(\mu ub\right)^2} \qquad (3) \qquad\qquad Q_S v_u + \frac{gh_u^2 b}{2} = Q_S v + \frac{g\left(h'\right)^2 b}{2} \qquad (4)$$

and the continuity equation $Q_S = vb\mu u = v_u h_u b$, with the flow under the sluice gate Q_S, the velocity before the hydraulic jump v, the velocity after the jump v_u, the upstream depth of flow h, the downstream depth before the jump h', the downstream depth behind the jump h_u, the width of the gate b, the width of the canal before the gate b_1 and the opening of the gate u (see figure 3). The coefficient μ depends on the geometry of the structure and the up– and downstream depths and is available from literature. In contrast to the standard models the velocity before the hydraulic jump v is calculated in such a way that there is a continuous passing from sub– to supercritical flow conditions.

Measurement results

In the following we compare measurement results at the plant with simulation results based on the developed model. For the simulation we take the measured flow through the turbines of the power stations Sill and Ruetz and the opening of the sluice gate as inputs, in addition the inflow to the supply canal at the sluice Matrei Q_{in} is assumed to be approximately constant. Due to the fact that the surge chamber Sill is modeled as a simple integrator and that the surge chamber Sill is relatively small, we have to adjust the model, because small variations in the inflow and the outflow of the surge chamber, respectively, caused by measurement errors in the flow through the turbine and errors in the model of the sluice gate, lead to big differences in the simulated water-level. Since this quantity has a big influence to the simulation, we have to guarantee that the deviation between the simulated water-level and the measured one is not to large. Therefore, the error between the calculated and the measured water level in the surge chamber Sill is fed to a PI-controller, to calculate the correction flow, which is necessary to guarantee that the simulated water level in the surge chamber Sill is approximately equal to the measured one. In figure 4, the measured and the calculated water-level at the sluice Matrei is shown. The differences between these two quantities, are mainly based on the estimation-error of the inflow Q_{in},

since the water-level depends on the integral of the difference between the inflow at the sluice and the outflow under the sluice gate over time and is therefore very sensitive to variations in the inflow Q_{in}. An additional reason especially for the differences after a time of 8 and 18 hours is the rough model of the sluice Matrei. Figure 5 depicts the water-level in the chamber before the sluice gate h and figure 7 shows the calculated flow under the sluice gate Q_S. The comparison of the measured water-level in the surge chamber Ruetz with the simulated one is depicted in figure 6. One should notice that the surge chamber Ruetz is modeled as an integrator and there are no corrections used for the simulation of this part of the model.

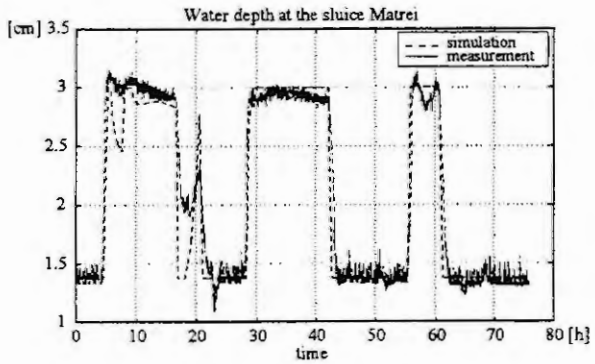


Figure 4: Water level at the sluice Matrei.

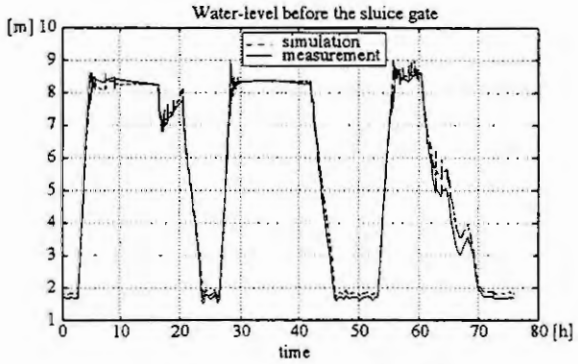


Figure 5: Water level before the sluice gate.

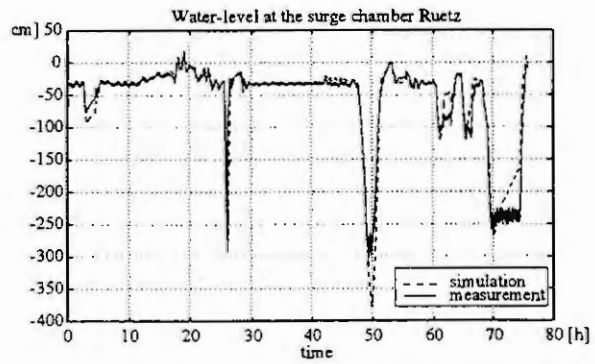


Figure 6: Water level in the surge chamber Ruetz.

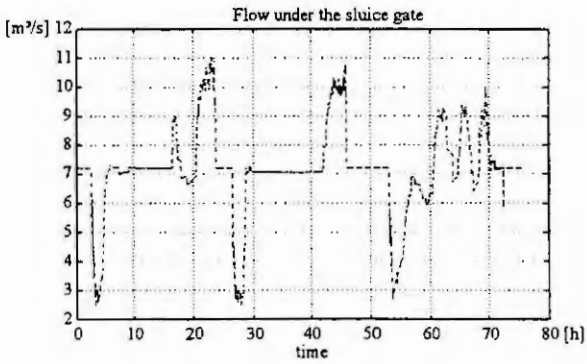


Figure 7: Flow under the sluice gate.

Conclusion

In this paper the mathematical model for the works water section of a power plant group is presented. The model is based on the nonlinear partial differential equations for the one-dimensional open channel and pipe flow and has to deal with a special coupling of open channel and pipe flow. With appropriate assumptions a significant model reduction is possible. Finally, measurement results at the plant show the excellent practical exactness of the proposed model.

References

1. Bollrich, G. and Preissler, G., Technische Hydrodynamik, volume 1, Verlag für Bauwesen, 1992.
2. Johnson, R. W., editor, Fluid Dynamics, CRC Press LLC, 1998.
3. Meusburger, M., Kugi, A., Schlacher, K., and Sillaber, A., Modeling and Control of a Special Type of Open Channel Flow, In: Proc. Computational Fluid Dynamics '98, Athens, Part I, volume 1, John Wiley & Sons Ltd., 1998, 292 – 297.
4. Mott, R. L., Applied Fluid Mechanics, Macmillan, 1994.
5. Truckenbrodt, E., Fluidmechanik, volume I and II, Springer - Verlag, 1989.
6. Wylie, E. B. and Streeter, V. L., Fluid Transients in Systems, Prentice Hall, 1993.

A LINEAR DECOUPLED MODEL OF OPEN-CHANNELS FOR THE SYNTHESIS OF A DECENTRALIZED VOLUME VARIATION OBSERVER

C. Seatzu, G. Usai
Department of Electrical and Electronic Engineering,
University of Cagliari, Piazza d'Armi — 09123 Cagliari, Italy

Abstract. In this paper we formulate a new linear decoupled dynamical model of open-channels. The main advantage is that it enables the design of a decentralized volume variation observer. In fact, open-channels require decentralization in the control. Valid solutions to this problem have been proposed in the literature. However, in all cases it has been assumed that state variables, i.e., volume variations in the pools, were available. Obviously, this is an unrealistic assumption. In this paper we propose the design of a decentralized volume variation observer that provides an estimation of the volume variation in each canal reach on the base of only local measurements, thus not vanishing the advantage of decentralization in the control.

1. Introduction

A wide variety of mathematical models describing the dynamic behaviour of open-channels have been proposed in the literature. Sometimes differences among them are really significant and involve the use of completely different control techniques.

In this paper we focus our attention on the problem of deriving a decoupled linear model of open-channels, such that a good estimation of volume variations with respect to a reference configuration of uniform flow, can be obtained on the base of only local measurements.

The model origins from a previous linear state-variable model of open-channels deduced by Corriga *et al.* in [1, 2]. The main advantage of the actual formulation is the decoupling between state space equations, that enables the design of an asymptotic state observer for each canal reach. In fact, open-channels, like all large scale systems, require decentralization in the control. Valid solutions to the problem of designing a decentralized constant-volume controller have been proposed in the literature [5, 6]. However, in all these cases an unrealistic assumption has been made. It has been assumed that state variables, i.e., volume variations in the pools, were available. Obviously, this is not the case in real applications. This paper just enables us to overcome this difficulty, not vanishing the advantage of decentralization in the control.

2. Linear decoupled dynamical model of the open-channels

Consider the system sketched in figure 1, consisting of a channel of n reaches joined by $n+1$ gates, where the last gate (the $(n+1)$-st) is fixed and the others are controlled. Let us suppose that water is conveyed to the first reach from a reservoir with constant level and that the level downstream from the final reach is also constant. All variables reported in figure 1, apart from those that define the geometry, represent the variations with respect to a reference configuration of uniform flow. It has been assumed that the canal cross section is trapezoidal. Its scheme is shown in figure 1.

The linear model of interest is derived from the Saint-Venant equations [7], thus the assumption of uniform flow as reference is essential. In particular, it origins from a previous centralized linear model of open-channels formulated by Corriga *et al.* [1, 2] and used by the authors for the synthesis of decentralized controllers [5, 6].

In this paper we limit to recall the main equations that are involved in the deduction of the new decoupled model. For more details on their derivation we address the reader to [1, 2]. Firstly,

$$q_{Ai}(t) = a_i\sigma_i(t) + b_i h_{Bi-1}(t) + c_i h_{Ai}(t), \qquad i = 1, \cdots, n, \tag{1}$$

where q_{Ai}, σ_i, and h_{Ai} denote the variations, with respect to the reference configuration, of the upstream discharge, the gate opening, and the upstream level of the i-th reach; h_{Bi-1} is the variation of the downstream level of the $(i-1)$-th reach, computed with respect to the same reference configuration; a_i, b_i and c_i, $i = 1, \cdots, n$, are constant values [2].

If we assume that no variation on users withdrawals occur, we can also write

$$q_{Bi}(t) = a_{i+1}\sigma_{i+1}(t) + b_{i+1}h_{Bi} + c_{i+1}h_{Ai+1}(t), \qquad i = 1, \cdots, n, \tag{2}$$

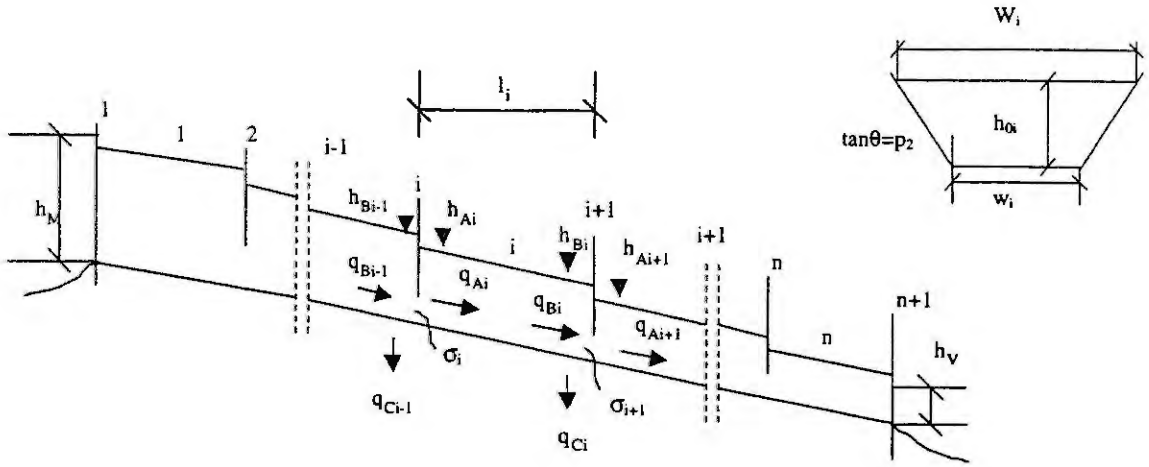


Figure 1: *Scheme of system composed by a cascade of n canal reaches and trapezoidal canal cross section.*

being $q_{Bi+1} = q_{Ai}$, $i = 1, \cdots, n-1$. Moreover, it has been proved that [1, 2]:

$$\begin{bmatrix} h_{Ai}(t) \\ h_{Bi}(t) \end{bmatrix} = \begin{bmatrix} K_{1i} \\ K_{3i} \end{bmatrix} v_i(t) + \begin{bmatrix} A'_{1i}(0) & A'_{2i}(0) \\ A'_{3i}(0) & A'_{4i}(0) \end{bmatrix} \begin{bmatrix} q_{Ai}(t) \\ q_{Bi}(t) \end{bmatrix}, \quad i = 1, \cdots, n, \tag{3}$$

where K_{1i}, K_{3i}, $A'_{ji}(0)$, $j = 1, \cdots, 4$, $i = 1, \cdots, n$, are appropriate constant values [2].

Substituting equations (1) and (2) into (3), we obtain the following matrix equation

$$\begin{bmatrix} h_{Ai}(t) \\ h_{Bi}(t) \end{bmatrix} = \overline{C}_i v_i(t) + \overline{D}_i \begin{bmatrix} \sigma_i(t) \\ \sigma_{i+1}(t) \\ h_{Ai+1}(t) \\ h_{Bi-1}(t) \end{bmatrix}, \qquad i = 1, \cdots, n, \tag{4}$$

where

$$\overline{C}_i = \begin{bmatrix} 1 - A'_{1i}(0)c_i & -A'_{2i}(0)b_{i+1} \\ -A'_{3i}(0)c_i & 1 - A'_{4i}(0)b_{i+1} \end{bmatrix}^{-1} \cdot \begin{bmatrix} K_{1i} \\ K_{3i} \end{bmatrix}, \tag{5}$$

$$\overline{D}_i = \begin{bmatrix} 1 - A'_{1i}(0)c_i & -A'_{2i}(0)b_{i+1} \\ -A'_{3i}(0)c_i & 1 - A'_{4i}(0)b_{i+1} \end{bmatrix}^{-1} \cdot \begin{bmatrix} A'_{1i}(0)a_i & A'_{2i}(0)a_{i+1} & A'_{2i}(0)c_{i+1} & A'_{1i}(0)b_i \\ A'_{3i}(0)a_i & A'_{4i}(0)a_{i+1} & A'_{4i}(0)c_{i+1} & A'_{3i}(0)b_i \end{bmatrix}. \tag{6}$$

Then, under the assumption of no external disturbance, it holds that

$$\dot{v}_i(t) = q_{Ai}(t) - q_{Bi}(t),$$

where q_{Ai} and q_{Bi} are defined in (1) and (2), respectively. Therefore, by taking into account the above equation (4), we get

$$\dot{v}_i(t) = \overline{A}_i v_i(t) + \overline{B}_i \begin{bmatrix} \sigma_i(t) \\ \sigma_{i+1}(t) \\ h_{Ai+1}(t) \\ h_{Bi-1}(t) \end{bmatrix} \tag{7}$$

where

$$\overline{A}_i = \begin{bmatrix} c_i & -b_{i+1} \end{bmatrix} \cdot \overline{C}_i, \tag{8}$$

$$\overline{B}_i = \begin{bmatrix} c_i & -b_{i+1} \end{bmatrix} \cdot \overline{D}_i + \begin{bmatrix} a_i & -a_{i+1} & -c_{i+1} & b_i \end{bmatrix}. \tag{9}$$

Now, let as define

$$\overline{u}_i(t) = \begin{bmatrix} \sigma_i(t) \\ \sigma_{i+1}(t) \\ h_{Ai+1}(t) \\ h_{Bi-1}(t) \end{bmatrix}, \quad \overline{y}_i(t) = \begin{bmatrix} h_{Ai}(t) \\ h_{Bi}(t) \end{bmatrix}.$$

In this way we get the n following linear independent systems:

$$\begin{cases} \dot{v}_i(t) = \overline{A}_i v_i(t) + \overline{B}_i(t)\overline{u}_i(t) \\ \overline{y}_i(t) = \overline{C}_i v_i(t) + \overline{D}_i(t)\overline{u}_i(t) \end{cases} \quad i = 1, \cdots, n. \tag{10}$$

The state variable of the i-th system is equal to the volume variation in the i-th reach; the inputs are the opening variations of the gates delimiting the pool (σ_i, σ_{i+1}), the upstream water level variation of the upstream gate (h_{Ai+1}) and the downstream water level variation of the downstream gate (h_{Bi-1}); finally, the output variables are equal to the upstream (h_{Ai}) and downstream (h_{Bi}) water level variation in the i-th reach.

3. Synthesis of the observer

In this section we briefly recall the main concepts relative to the design of an asymptotic state observer for linear time–invariant systems [3].

Let us consider a linear time–invariant system in state space form:

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) \end{cases} \tag{11}$$

where $\boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{u} \in \mathbb{R}^m$, $\boldsymbol{y} \in \mathbb{R}^p$ are the state, the input and the output vector, respectively; $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, and $\boldsymbol{D}$ are constant matrices of appropriate size.

If the state is not directly available for measurement, a well known approach is to construct an observer of the form

$$\begin{cases} \dot{\hat{\boldsymbol{x}}}(t) = \boldsymbol{A}\hat{\boldsymbol{x}}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \boldsymbol{G}[\boldsymbol{y}(t) - \hat{\boldsymbol{y}}(t)] \\ \hat{\boldsymbol{y}}(t) = \boldsymbol{C}\hat{\boldsymbol{x}}(t) + \boldsymbol{D}\boldsymbol{u}(t) \end{cases} \tag{12}$$

where the constant matrix $\boldsymbol{G}$ weights the errors on the output entries. If we define

$$\boldsymbol{e}(t) = \boldsymbol{x}(t) - \hat{\boldsymbol{x}}(t)$$

then $\boldsymbol{e}(t)$ represents the state reconstruction error at the time instant t. It is easy to prove that its dynamic is governed by the following linear equation

$$\dot{\boldsymbol{e}}(t) = [\boldsymbol{A} - \boldsymbol{G}\boldsymbol{C}]\boldsymbol{e}(t). \tag{13}$$

Obviously, the effectiveness of the above observer depends on the choice of the constant matrix $\boldsymbol{G}$. A lot of procedures can be applied to compute an appropriate matrix $\boldsymbol{G}$. In particular, $\boldsymbol{G}$ can be chosen so as to impose the desired closed–loop eigenvalues to the matrix $\boldsymbol{A} - \boldsymbol{G}\boldsymbol{C}$.

At this point it immediately appears the usefulness of the linear decoupled model of open–channels formulated in the previous section. In fact, the n independent systems are in the form required for the application of the theory above. In the case at hand, we will have to compute n different vectors $\boldsymbol{G}$, one for each canal reach. In this way each observer only uses local informations, thus not vanishing the advantage of decentralization in the controller.

4. An applicative example

Let us consider a two-reach canal, corresponding to the general scheme shown in figure 1, with the following characteristics: length of the first reach: $l_1 = 4000m$; length of the second reach: $l_2 = 5000m$; canal bottom slope: $p_1 = 0.0003$; water level depth in upstream reservoir in reference to the canal bottom in the upper end section: $h_M = 2.5m$; water level depth in downstream reservoir in reference to the canal bottom in the lower end section: $h_V = 1m$; trapezoidal cross section (see figure 1) with $w = 1.7m$, $\theta = 45°$; constant opening section of the third gate: $\lambda_3 = \lambda_{03} = 2.41m^2$; discharge coefficient: $\eta = 0.6$; roughness coefficient: $\gamma = 0.36$.

The nominal configuration of uniform flow is characterized by the following levels and discharge values: water level depth in the 1-st reach: $h_{01} = 1.70m$; water level depth in the 2-nd reach: $h_{02} = 1.20m$; flow rate in the 1-st reach : $q_{01} = 5.94m^3/s$; flow rate in the 2-nd reach : $q_{02} = 3.02m^3/s$; user flow rate at the 1-st reach lower end: $q_{0c1} = 2.92m^3/s$; user flow rate at the 2-nd reach lower end: $q_{0c2} = 2.50m^3/s$; opening section of the 1-st gate: $\lambda_{01} = 2.50m^3/s$; opening section of the 2-nd gate: $\lambda_{02} = 1.61m^3/s$.

In this case the two linear systems used for the synthesis of the observers are characterized by the following constant matrices:

$$\overline{A}_1 = -2.1927 \cdot 10^{-4}, \quad \overline{A}_2 = -2.4738 \cdot 10^{-4},$$

$$\overline{B}_1 = [\ 1.9688 \quad -1.4314 \quad 2.1156 \quad 3.0450\ ], \quad \overline{B}_2 = [\ 1.4445 \quad -0.3550 \quad -5.1702 \quad 2.1349\ ],$$

$$\overline{C}_1 = \begin{bmatrix} 0.3289 \\ 0.4607 \end{bmatrix} \cdot 10^{-4}, \quad \overline{C}_2 = \begin{bmatrix} 0.2369 \\ 0.2536 \end{bmatrix} \cdot 10^{-4},$$

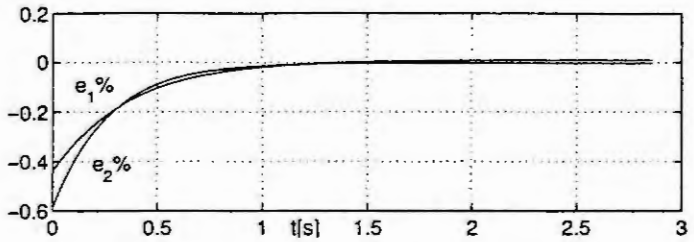


Figure 2: *The results of numerical simulation.*

$$\overline{D}_1 = \begin{bmatrix} 0.2636 & 0.0703 & -0.1039 & 0.4077 \\ -0.1078 & -0.2315 & 0.3421 & -0.1667 \end{bmatrix}, \quad \overline{D}_2 = \begin{bmatrix} 0.3380 & 0.0205 & 1.5991 & 0.4971 \\ -0.0698 & -0.1243 & 0.0205 & -0.1131 \end{bmatrix}.$$

By taking into account the numerical values of $\overline{A}_i$ and $\overline{C}_i$, $i = 1, 2$, it is easy to prove that stability holds for whatever couple of gain vectors $\overline{G}_i$, $i = 1, 2$, whose entries are positive real numbers.

Note that, by virtue of the dimensions of $\overline{A}_i$, $\overline{C}_i$ and $\overline{G}_i$ the same eigenvalues for $\overline{A}_i - \overline{G}_i\overline{C}_i$, $i = 1, 2$, can be assigned by an infinite number of gain vectors $\overline{G}_i$. A satisfactory solution has been determined by a trial and error procedure. In particular, it has been evaluated that a good choice is

$$\overline{G}_1 = [\ 0.8 \quad 0.4\ ], \quad \overline{G}_2 = [\ 2.8 \quad 2\ ],$$

that corresponds to $\lambda_1 = -2.6401 \cdot 10^{-4}$ and $\lambda_2 = -3.6443 \cdot 10^{-4}$ as eigenvalues of $\overline{A}_1 - \overline{G}_1\overline{C}_1$ and $\overline{A}_2 - \overline{G}_2\overline{C}_2$, respectively.

Note that the above numerical values have been obtained as a consequence of many simulation test cases, where also the presence of a decentralized controller is taken into account. The results of some of these simulations are reported in [6]. In this paper we limit to consider the open–loop system and an initial error on the state estimate. We assume that no regulator is acting on the net and no variation on users withdrawals occurs. The results of simulation are reported in figure 2 which presents the percentage errors on the state estimation. As it can be seen a null steady state error is obtained.

Note that simulation has been carried out using the commercial SIC software, a completely nonlinear model of open–channels developed at Cemgref (Montpellier, France) [4].

5. Conclusions

In this paper the formulation of a linear decoupled dynamical model of open–channels has been proposed. It origins from the Saint–Venant equations, linearized around a reference configuration of uniform flow and uses some physical equations firstly derived by Corriga *et al.* in previous works. The importance of the new formulation is that it enables the synthesis of a decentralized state observer, that is capable of reconstructing the volume variation in each canal reach on the base of only local measurements. This is an important problem to be solved since the control of open–channels is always decentralized and a state observer is an essential requirement in real applications.

References

[1] G. Corriga, S. Sanna, G. Usai, Sub-optimal constant volume control for open-channel networks, *Appl. Math. Modelling*, pp. 262–267, July 1983.

[2] G. Corriga, S. Sanna, G. Usai, Estimation of uncertainty in an open-channel network mathematical model, *Appl. Math. Modelling*, pp. 651–657, November 1989.

[3] H. Kwakernaak, R. Sivan, Linear Optimal Control Systems, Wiley Interscience (New York), 1972.

[4] P.O. Malaterre, J.P. Baume, SIC 3.0, a simulation model for canal automation design, *Proc. Int. Workshop on Regulation of irrigation canals*, Marrakech, April 1997.

[5] C. Seatzu, "Design and robustness analysis of decentralized constant volume-control for open-channels," *Applied Mathematical Modelling*, 1999.

[6] C. Seatzu, "Decentralized control of irrigation open–channels via eigenstructure assignment," *3rd IMACS MathMod*, Vienna (Austri), 2000.

[7] T. Weiyan, *Shallow Water Hydrodynamics*, Elsevier, Amsterdam, The Netherlands, 1992.

DECENTRALIZED CONTROL OF IRRIGATION OPEN–CHANNELS VIA EIGENSTRUCTURE ASSIGNMENT

C. Seatzu
Department of Electrical and Electronic Engineering,
University of Cagliari, Piazza d'Armi — 09123 Cagliari, Italy

Abstract. In this paper we examine the problem of designing a decentralized constant–volume controller for open–channel hydraulic systems. The system dynamic is described by a linear, time–invariant model deduced from the Saint–Venant equations by Corriga *et al.* in previous works. The synthesis procedure allows us to derive a parametric expression for the set of feedback gains which achieve the desired eigenvalue assignment. The free parameters in this parametric expression can be used to assign eigenvectors as close to the desired ones as possible, while achieving the required eigenvalue assignment.

1. Introduction

Over the last decades, much research effort has been devoted to water flow control of open–channel conveyance systems such as irrigation channels. A great number of regulatory procedures have been proposed.

In this paper we consider a mathematical model of open-channels firstly proposed by Corriga *et al.* in [2] that expresses the dynamic relationships, in terms of transcendental functions, between the gate opening sections and the corresponding stored water volume variations in the different canal reaches with respect to an initial reference configuration of uniform flow. Series expansion around $s = 0$ gives a state variable linear and time invariant model that enables us to design an efficient decentralized constant–volume control law. In this way the stored volumes in the different reaches can be maintained practically constant, even with variations in users withdrawals, by acting only on the upstream gate of the reach whose volume variation is detected. Thus decentralization, i.e., one of the main requirements for large–scale systems like the hydraulic one herein examined, is satisfied.

The synthesis procedure followed in this paper has been firstly proposed by Lu *et al.* in [3]. It enables us to derive a parametric expression for the set of feedback gains of the decentralized controller which achieve the desired eigenvalue assignment. The free parameters in this parametric expression can be used to assign eigenvectors as close to the desired ones as possible while achieving the required eigenvalue assignment. This requirement arises from the fact that the speed of the dynamic response of a linear system depends on its eigenvalues whereas the "relative shape" of the dynamic response depends on the associated eigenvectors. Over the past decade, many methods have been proposed for pole placement by decentralized control [1, 8]. However, it is not easy to incorporate eigenvector assignment into the process of eigenvalue assignment using these methods.

The results of some numerical simulations will be reported in the final paper. All of them have been carried out on the commercial SIC software, a completely nonlinear model of open–channels developed at Cemagref (Montpellier, France) [5] and show that satisfactory results can be obtained when the above control law is implemented. Note that it has been also assumed that the state is not directly measured but is reconstructed by means of the asymptotic state observer proposed by the author in [7].

2. Linear approximate model

Consider the system sketched in Figure 1 of [7], consisting of a channel of n reaches joined by $n+1$ gates, where the last gate is fixed and the others are controlled. Let us suppose that water is conveyed to the first reach from a reservoir with constant level and that the level downstream from the final reach is also constant. All the variables considered, apart from those that define the geometry, represent the variations with respect to a reference configuration of uniform flow in each reach. In particular, let $\boldsymbol{v} = [v_1, \cdots, v_i, \cdots, v_n]^T$, $\sigma = [\sigma_1, \cdots, \sigma_i, \cdots, \sigma_n]^T$, $\boldsymbol{q_C} = [q_{C1}, \cdots, q_{Ci}, \cdots, q_{Cn}]^T$, where v_i, σ_i and q_{Ci} are the stored volume variation in the i–th reach, the variation in the i–th gate opening section and the user flow variation at the i–th reach lower end, respectively.

The linear model presented in [2] and used in this paper for the synthesis of the decentralized controller, has been derived by first linearizing the Saint-Venant equations for the unsteady flow of water in open-channels [2] around a reference condition of uniform flow. Then, since the obtained equations are linear, the Laplace transform technique, with appropriate initial and boundary conditions, has been used to solve them. Since the model needs to be accurate in the low-frequency range, where the most significant phenomena take place, $s = j\omega = 0$ is taken as initial point and the series expansion may be truncated to the second term. Finally, by inverse L–transforming, a linear time–invariant model with the following structure can be obtained:

$$\dot{\boldsymbol{v}}(t) = \boldsymbol{A}\boldsymbol{v}(t) + \boldsymbol{B}\sigma(t) \tag{1}$$

where $\boldsymbol{A}$ and $\boldsymbol{B}$ are constant matrices [2]. Finally, taking into account the variations of the users flow rates $\boldsymbol{q_C}$, equation (1) can be rewritten as:

$$\dot{\boldsymbol{v}}(t) = \boldsymbol{A}\boldsymbol{v}(t) + \boldsymbol{B}\sigma(t) - \boldsymbol{I}\boldsymbol{q_C}(t) \tag{2}$$

where $\boldsymbol{I}$ is the n order identity matrix.

For more details on the construction of the mentioned linear model we address to [2].

3. Eigenstructure assignment by decentralized feedback control

In this section we recall the fundamental steps of an efficient method proposed by Lu *et al.* [3] to design a decentralized control law via eigenstructure assignment.

Consider a linear time-invariant system described in state space

$$\dot{x}(t) = Ax(t) + \sum_{i=1}^{\nu} B_i u_i(t), \quad y_i(t) = C_i x(t), \quad i = 1, \cdots, \nu \tag{3}$$

where $x(t) \in \mathcal{R}^n$ is the state, $u_i(t) \in \mathcal{R}^{m_i}$, $y_i(t) \in \mathcal{R}^{r_i}$ are the input and output respectively, of the i–th local control station. Matrices A, B_i and C_i, $i = 1, \cdots, \nu$ are real, constant and of appropriate size.

If a decentralized feedback control law $u_i(t) = F_i y_i(t)$, $F_i \in \mathcal{R}^{m_i \times r_i}$, $i = 1, \cdots, \nu$ is applied to system (3), a closed-loop system is obtained in the form

$$\dot{x}(t) = \left(A + \sum_{i=1}^{\nu} B_i F_i C_i \right) x(t) = (A + BFC)x(t) \tag{4}$$

where

$$B = [\ B_1 \quad \cdots \quad B_\nu\ ] \in \mathcal{R}^{n \times m}, \quad m = \sum_{i=1}^{\nu} m_i \tag{5}$$

$$C = \begin{bmatrix} C_1 \\ \vdots \\ C_\nu \end{bmatrix} \in \mathcal{R}^{r \times n}, \quad r = \sum_{i=1}^{\nu} r_i \tag{6}$$

$$F = \text{Block diag}[\ F_1, \quad \cdots \quad F_\nu\ ] \in \mathcal{R}^{m \times r}. \tag{7}$$

Let us define the set of decentralized feedback gains by

$$\mathcal{F} = \left\{ F = \text{Block diag}[\ F_1, \quad \cdots \quad F_\nu\ ] \in \mathcal{R}^{m \times r} : F_i \in \mathcal{R}^{m_i \times r_i} \right\}. \tag{8}$$

Lu *et al.* in [3] provide a valid procedure to determine a decentralized feedback gain matrix $F \in \mathcal{F}$ so that

1. (*Eigenvalue Assignment*): System (4) is assigned an arbitrary self-conjugate set of k eigenvalues $\{\mu_i, i = 1, \cdots, k\}, k \leq n$, namely

$$\{\mu_i, i = 1, \cdots, k\} \subset \Lambda(A + BFC) = \Lambda\left(A + \sum_{i=1}^{n} B_i F_i C_i \right) \tag{9}$$

   where $\Lambda(\ )$ denotes the set of eigenvalues.

2. (*Eigenvector Assignment*): $v_i, i = 1, \cdots k$, the closed-loop eigenvectors (with unit norm) associated with $\{\mu_i, i = 1, \cdots, k\}$, minimize certain performance indexes.

One commonly used performance index for assigning eigenvectors $v_i, i = 1, \cdots, k$ is

$$J = \sum_{i=1}^{k} (v_i - v_i^d)^T W (v_i - v_i^d) \tag{10}$$

where $W = \text{diag}(w_{i1}, \cdots, w_{in})$ is a weighting matrix, $\{v_i^d, i = 1, \cdots, k\}$ is a set of desired unit normed eigenvectors which reflect our requirement on the shape of the closed-loop dynamic response. In general, it is not possible to exactly assign v_i to v_i^d and minimization of (10) gives the eigenvectors closest to the desired ones [3].

3.1. Eigenvalue assignment

In this subsection, we recall several analytical results firstly proposed and demonstrated by Lu *et al.* in [3], to characterize the decentralized feedback gain matrices that achieve the desired eigenvalue assignment for linear system (3). Proofs are omitted here as they have been derived in [3] and we address to it for more details. Furthermore, in the rest of the note we assume that linear system (3) is a controllable and observable system.

Theorem 1. *(Characterization of Decentralized Control for Eigenvalue Assignment)*: Let $\{\mu_i, i = 1, \cdots, k\}$ be a self-conjugate set of distinct complex numbers such that $\{\mu_i, i = 1, \cdots, k\} \cap \Lambda(A) = \emptyset$.

There exists a decentralized feedback gain $F \subset \mathcal{F}$ such that the eigenvalues of the closed-loop system (4) contain $\{\mu_i, i = 1, \cdots, k\}$ *if and only if* there exist non-null parameter vectors $p_i \in \mathcal{C}^m, i = 1, \cdots, k$, satisfying the following two conditions:

1. $p_i \in \mathcal{R}^m$ if μ_i is real, and $p_i = p_j^* \in \mathcal{C}^m$ if $\mu_i = \mu_j^*$;

2.
$$[I - FH(\mu_i)]\, p_i = 0, \quad i = 1, \cdots, k \tag{11}$$

where $H(s) = C(sI - A)^{-1}B$ is the open-loop transfer function. ■

Theorem 2. *(Existence of Decentralized Control)*: Assume that the eigenvalues of A are distinct. Let $\{\lambda_i, i = 1, \cdots, k\} \subset \Lambda(A)$ and $v_i, w_i, i = 1, \cdots, k$ be the corresponding right and left eigenvectors of A. Define a matrix T by

$$T = \begin{bmatrix} (C_1 v_1)^T \otimes (w_1^T B_1) & \cdots & (C_1 v_\nu)^T \otimes (w_\nu^T B_1) \\ \vdots & \ddots & \vdots \\ (C_1 v_k)^T \otimes (w_k^T B_1) & \cdots & (C_\nu v_k)^T \otimes (w_k^T B_\nu) \end{bmatrix} \tag{12}$$

where $M \otimes V$ is the Kronecker product of M and V.

If rank(T)= k, then there exists an $\epsilon > 0$ such that for any set $\{\mu_i, i = 1, \cdots, k\}$ satisfying

$$\mu_i \in \mathcal{D}_i \equiv \{\mu : 0 < |\mu - \lambda_i| \leq \epsilon\}, \quad i = 1, \cdots, k$$

there exists a solution $F \in \mathcal{F}$ to

$$[I - FH(\mu_i)]\, p_i = 0, \quad i = 1, \cdots, k \tag{13}$$

where $p_i \neq 0$, $i = 1, \cdots, k$. ■

By taking into account the above theorems, Lu *et al.* [3] provide a parametric expression for a decentralized feedback gain matrix.

The following notations are useful in stating the next theorem:

$$H_i(s) = C_i(sI - A)^{-1}B \in \mathcal{C}^{r_i \times m}, \quad I_{m\times m} = \begin{bmatrix} E_1 \\ E_2 \\ \vdots \\ E_\nu \end{bmatrix} \begin{matrix} \}m_1 \\ \}m_2 \\ \vdots \\ \}m_\nu \end{matrix} \quad m = \sum_{i=1}^{\nu} m_i$$

$$M^+ = \begin{cases} M^T(MM^T)^{-1}, & \text{if } M \in \mathcal{C}^{p\times q}, p < q \\ (M^TM)^{-1}M^T, & \text{if } M \in \mathcal{C}^{p\times q}, p \geq q \end{cases}$$

$$M^\perp = \begin{cases} I - M^T(MM^T)^{-1}M, & \text{if } M \in \mathcal{C}^{p\times q}, p < q \\ 0 \in \mathcal{C}^{q\times q}, & \text{if } M \in \mathcal{C}^{p\times q}, p \geq q \end{cases}$$

where $I_{m\times m}$ is an m dimensional identity matrix.

Theorem 3. *(A Parametric Expression for Decentralized Control)*:

Let $\{\mu_i = 1, \cdots, k\}$ be a self-conjugate set of distinct complex numbers such that $\{\mu_i = 1, \cdots, k\} \cap \Lambda(A) = \emptyset$. Then the eigenvalues of the closed-loop system (4) contains $\{\mu_i = 1, \cdots, k\}$ if

$$F_i = E_i[p_1, \cdots, p_k][H_i(\mu_1)p_1, \cdots, H_i(\mu_k)p_k]^+, \quad i = 1, \cdots, \nu \tag{14}$$

where p_i, $i = 1, \cdots, k$ satisfy

1. $p_i \in \mathcal{R}^m$ if μ_i is real, and $p_i = p_j^* \in \mathcal{R}^m$ if $\mu_i = \mu_j^*$;

2. matrix $[p_1, \cdots, p_k][H_i(\mu_1)p_1, \cdots, H_i(\mu_k)p_k] \in \mathcal{C}^{r_i \times k}$ is of full rank, $i = 1, \cdots, \nu$;

3.
$$E_i[p_1, \cdots, p_k][H_i(\mu_1)p_1, \cdots, H_i(\mu_k)p_k]^\perp = 0, \quad i = 1, \cdots, \mu \tag{15}$$
$$\| p_j \| = 1, \quad j = 1, \cdots, k, \tag{16}$$

where $\| \ \|$ is the euclidean norm. ■

Two main remarks can be done. First of all, the vector p_i explicitly parameterizes the closed-loop right eigenvector v_i associated with μ_i according to

$$v_i = (\mu_i I - A)^{-1}Bp_i. \tag{17}$$

Furthermore, equation (14) gives a parametric expression of the desired feedback gain F_i, where the parameter vectors $p_i \in \mathcal{C}^m$, $i = 1, \cdots, k$, satisfy matrix nonlinear equations (15) and (16). If $r_i \geq k$, then $[H_i(\mu_1)p_1, \cdots, H_i(\mu_k)p_k]^\perp = 0$, therefore, (15) is trivially satisfied for any set of p_i with $\| p_i \| = 1$. On the other hand, if $r_i < k$, then the matrix nonlinear equation (15) gives $m_i k$ nonlinear equations corresponding to this r_i. In order to obtain a set of p_i, $i = 1, \cdots, k$ satisfying (15), one needs to solve at most $\sum_{i=1}^{\nu} m_i k$ nonlinear equations. At this purpose, standard methods for solving nonlinear equations, such as Newton's method can be used.

3.2. Eigenstructure assignment

Equation (14) provides a parametric expression for a class of decentralized feedback gains that achieve the desired eigenvalue assignment. In order to find such a feedback gain, we need to determine p_i, $i = 1, \cdots, k$ satisfying conditions 1), 2) and 3) stated in Theorem 3. In many cases, p_i, $i = 1, \cdots, k$ satisfying those conditions are nonunique. This freedom allows us to achieve eigenvector assignment in addition to eigenvalue assignment. For example, we may consider the following eigenstructure problem: find the parameter vectors p_i, $i = 1, \cdots, k$ which achieve the following minimization

$$\min_{p_i, i=1,\cdots,k} \sum_{i=1}^{k} (v_i(p_i) - v_i^d)^T W (v_i(p_i) - v_i^d) \tag{18}$$

subject to

$$E_i[p_1, \cdots, p_k][H_i(\mu_1)p_1, \cdots, H_i(\mu_k)p_k]^{\perp} = 0, \quad i = 1, \cdots, \nu \tag{19}$$

$$\| v_i(p_i) \| = 1, \quad i = 1, \cdots, k \tag{20}$$

where $v_i = (\mu_i I - A)^{-1} B p_i$.

The above minimization problem attempts to assign eigenvectors as close to the desired ones as possible while achieving required eigenvalue assignment as guaranteed by constraints (19) and (20). Constraint (20) ensures the unit norm of eigenvectors v_i, $i = 1, \cdots, k$. It is equivalent to constraint (16) in Theorem 3 in the sense that it ensures the nonzeroness of eigenvectors v_i, $i = 1, \cdots, k$.

When the minimization problem is solved, the required decentralized feedback gain matrix is obtained by substituting the resulting p_i, $i = 1, \cdots, k$ into formula (14).

4. An applicative example

In this subsection we consider the same test canal already used in [7] and show how the above procedure can be satisfactorily applied to design a decentralized state feedback control law, i.e., how it is possible to determine a diagonal feedback gain matrix F such that the closed-loop behaviour of system (1) is characterized by the desired set of eigenvalues and the eigenvectors are as close as possible to the desired ones. Since we want to design a state feedback control law, matrix C has been chosen equal to the two order identity matrix.

In our numerical example, in accordance with the notation used in Section 3, we have $n = m = r = \nu = 2$, $m_i = r_i = 1$, $i = 1, 2$. That is, we assume that the control actions are the gate opening variations and each one is a function of the only volume variation in the downstream gate, assumed to be measurable.

First of all, it has been verified that the system is controllable and the rank condition required by Theorem 2 is satisfied.

Furthermore, a satisfactory closed-loop dynamic has been imposed. As already specified in the Introduction, in previous works the above linear model has been used to design a centralized control law by means of an LQR technique [2], i.e., the constant feedback control law has been determined by solving an algebraic Riccati equation [4]. It has been assumed that the performance index to be minimized is

$$J = \int_0^{\infty} [x^T(t) Q x(t) + u^T(t) R u(t)] dt$$

where

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & v_{10}/v_{20} \end{bmatrix}, \quad R = 50000.$$

The structure of Q has been chosen such that the volume variations in each reach, with respect to the initial volume, are weighted in the same manner. R has been assumed to be scalar so as to control all the gates with the same energy. $R = 50000$ is an appropriate numerical value determined in [2] by a trial and error procedure.

In this paper we consider the resulting closed-loop behaviour as a target, i.e., we assume that the desired eigenstructure is that obtained when the above mentioned centralized control law is applied. Furthermore, we chose $W = I$ in equation (18) as we want to equally penalize the two distances between the desired and the actual eigenvectors.

In our numerical example, the constrained minimization problem (18-19-20) has been solved by means of the routine `constr` of MATLAB. Such a routine requires an initial estimate of the solution. It has been evaluated that good initial values for p_i, $i = 1, 2$ can be obtained by assuming $v_i = v_i^d$, $i = 1, 2$ in equation (17).

Satisfactory results have been obtained: the eigenvalues are exactly coincident with the desired ones and only minor errors on the first eigenvector occur.

Now, let us present the result of a numerical simulation carried out on the completely nonlinear model of open-channels developed at Cemagref (Montpellier, France) [5]. Let us assume step disturbances acting on both canal reaches after a time period equal to 20'. In particular, we consider $q_{C1} = -0.3 m/s^3$ and $q_{C2} = -0.25 m/s^3$. Let us also assume that the volume variation observer proposed in [7] is used to reconstruct the system state. Note that in this case a proportional action has been added to the observer gains so as to obtain a null steady state error even in presence of unknown step disturbances. In particular,

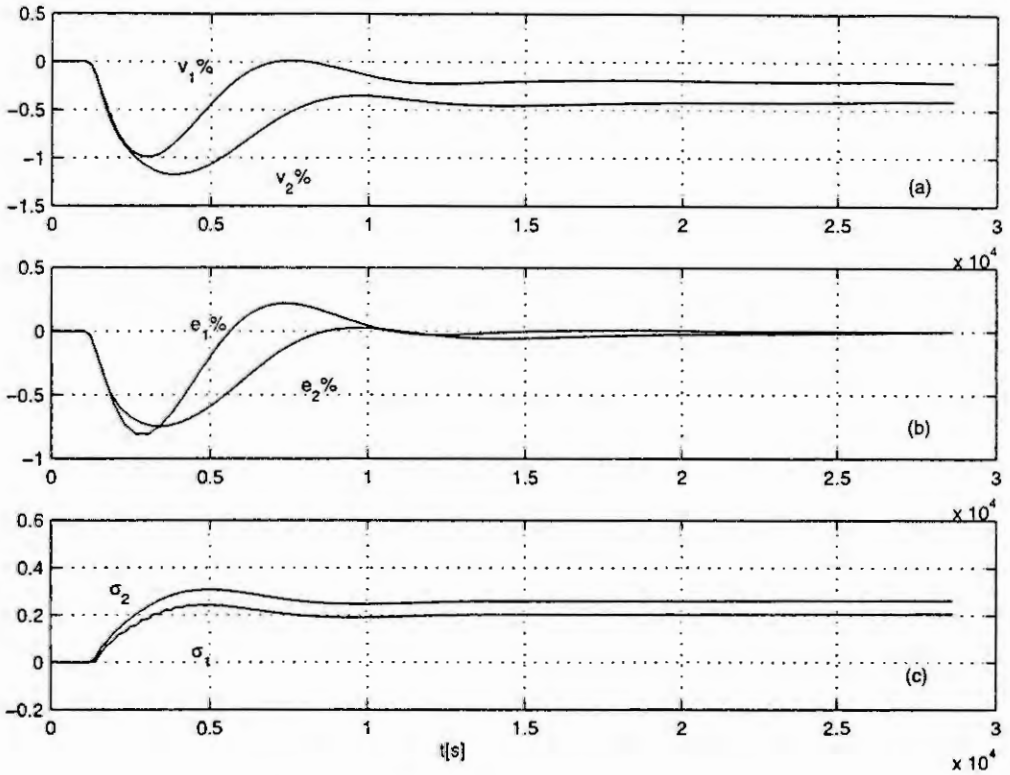


Figure 1: *The results of numerical simulation.*

let $\overline{G}_1 = [0.8 + 3 \cdot 10^{-6}/s,\ 0.4 + 3 \cdot 10^{-6}/s]$, $\overline{G}_2 = [2.8 + 5 \cdot 10^{-6}/s,\ 2 + 4 \cdot 10^{-6}/s]$, where the above values have been determined via a trial and error procedure.

Results of the numerical simulation are presented in figure 1: (a) shows the volume percentage variations; (b) the percentage errors on the estimate; (c) the control actions, i.e., the gate opening variations. As it can be seen a null steady state error is obtained even in the presence of unknown users withdrawals.

5. Conclusions

In this paper the problem of designing a decentralized constant volume controller for open–channels has been investigated. In particular, a procedure that enables to design the decentralized feedback gains so as to impose the desired structure to the closed–loop system, has been satisfactorily applied.

The results of a numerical simulation carried out on a completely nonlinear model of open–channels, has also been presented. Note that it has been assumed that the state is not directly measured, but it is reconstructed by the decentralized state estimator firstly proposed by the author in [7].

References

[1] J.H. Chow, M.A. Kale, "Decentralized local pole-placement control designs using static output feedback," *Proc. 1989 Amer. Contr. Conf.*, pp. 236-241, June 1989.

[2] G. Corriga, S. Sanna, G. Usai, "Sub-optimal constant volume control for open-channel networks," *Appl. Math. Modelling*, pp. 262–267, July 1983.

[3] J. Lu, H. D. Chiang and J.S. Thorp, "Eigenstructure Assignment by Decentralized Feedback Control," *IEEE Tran. on Automatic Control*, Vol. 38, No. 4, pp. 587–594, April 1993.

[4] H. Kwakernaak, R. Sivan, Linear Optimal Control Systems, Wiley Interscience (New York), 1972.

[5] P.O. Malaterre, J.P. Baume, "SIC 3.0, a simulation model for canal automation design," *Proc. Int. Workshop on Regulation of irrigation canals*, Marrakech, (1997).

[6] C. Seatzu, "Design and robustness analysis of decentralized constant volume–control for open–channels," *Appl. Math. Modelling*, pp. 479–500, June 1999.

[7] C. Seatzu, G.Usai, "A linear decoupled model of open–channels for the synthesis of a decentralized volume variation observer," *3rd IMACS MathMod*, Vienna (Austri), 2000.

[8] M. Tarokh, "Approach to pole assignment by centralized and decentralized output feedback," *IEE Proc.* Vol. 136, No. 2, pp. 89–97, March 1989.

A SOTWARE ENVIRONMENT FOR THE SIMULATION AND THE CONTROL LAW DESIGN OF THE *SCIROCCO* PLASMA WIND TUNNEL

G. Ambrosino[1] and M. Mattei[2]
[1]Dipartimento di Informatica e Sistemistica, Università degli Studi di Napoli Federico II
Via Claudio 21, 80125, Napoli, ITALY
[2]Dipartimento di Informatica, Matematica, Elettronica e Trasporti, Università degli Studi di Reggio Calabria
Via Graziella, Località Feo di Vito, 89100 Reggio Calabria, ITALY

Abstract. In this paper a software environment for the simulation and the control law design of the *Scirocco* Plasma Wind Tunnel under construction at CIRA (*Italian Aerospace Research Center*) is presented. A first version of the PWT DSS (*Dynamic Simulation Software*) was a control design oriented simplified dynamic simulator of the plant which was built to carry out a process-control integrated design. To give an answer to the increasing questions of the project engineers of the different plant subsystems during the design phase this SW was gradually enriched to become a complete simulation environment. The main function of the DSS during the operating life of the facility will be to support the test engineer in the choice of the plant configuration and in the control law design for the tracking of desired stagnation temperature and pressure time trajectories. As for the control law a *feedforward* plus feedback control strategy has been chosen: the *feedforward* action is computed *off-line* on the basis of the nonlinear plant model, while the feedback control action is obtained by means of a gain scheduled PID designed on the basis of a linear parameter varying model of the plant.

Introduction and Plant Description

When dealing with complex plants, it is often required to control engineers to build a dynamic simulation model for the synthesis of the control laws. In the case that the plant is already existing or other similar plants are already working, one way to obtain such a model is to *identify* it on the basis of experimental tests. Several techniques are available to this end which make use of more or less *know how* on the physical process (see for example [8]). On the other hand, in the case that the plant we deal with has never been built before and it is too costly to build prototypes, a great effort has to be spent to build the dynamic simulation model of the plant on the basis of the knowledge of the physical phenomena arising during the plant operating. At a later stage, after the plant building, the model can be *calibrated* on the data available from real measurements.

This is the case of the *Scirocco Plasma Wind Tunnel* (PWT), which is now into its realization phase at CIRA and is to be the largest arc heated facility ever built for studying the re-entry vehicle thermal protection system by means of experimental tests. The above facility will permit to study the effects of the aerothermodynamic phenomena arising on some parts of space vehicles during the re-entry phase, by reproducing, on suitable test models, proper thermal stress conditions specified in terms of pressure and temperature time trajectories.

A schematic of the *Scirocco* PWT is depicted in Figure 1. The core of the plant is the test chamber; it connects the nozzle with the diffuser and houses the test model and its support system. The high level of enthalpy required into the aerodynamic flow is assured by the electric arc heater fed by the compressed air system through an instrumented pipe line which allows for the regulation of the mass flow. The electrical power to the arc heater is supplied through a 70 MW power supply system. From the arc heater the plasma flow is accelerated by the nozzle up to the required velocity. Downstream the test chamber, the pressure is recovered by the diffuser, in order to reduce the size of the vacuum system. The heat exchanger allows the high enthalpy air flow energy recovery. To automatically execute the experiments, the PWT will be endowed with an Automation System consisting of various *Local Control Systems* (LCSs) and a higher level *Supervisory System*. The Supervisor will act as a sort of virtual operator; it will receive data and messages from the LCSs, and will send them commands and set points for the local regulation loops. In particular, in order to guarantee the tracking of the desired pressure and temperature time trajectories on the test article, the Supervisor will use a *Test Control Module* (TCM) which is entrusted with the on-line automatic generation of the set points for the electric power supply, the air compressed and the vacuum regulation systems.

The objective of this paper is to describe a simulation software environment, named PWT DSS (*Dynamic Simulation Software*), which has been developed to carry out a process-control integrated design of the plant and to give a concrete aid to the project engineers of all the relevant subsystems of the plant. Indeed, as confirmed by our experience, the project engineer of one single subsystem of a complex plant often needs to have information about the behavior of the interacting subsystems. Moreover, in the case of *Scirocco* which is to be the first PWT where *time trajectories* in temperature and pressure on the test model have to be reproduced, the dynamic

simulator was a fundamental tool to put in evidence all the potentially critical situations due to the *dynamic interaction* of the different plant subsystems. In facts, in a preliminary design phase all the subsystems were dimensioned referring to *static* test conditions.

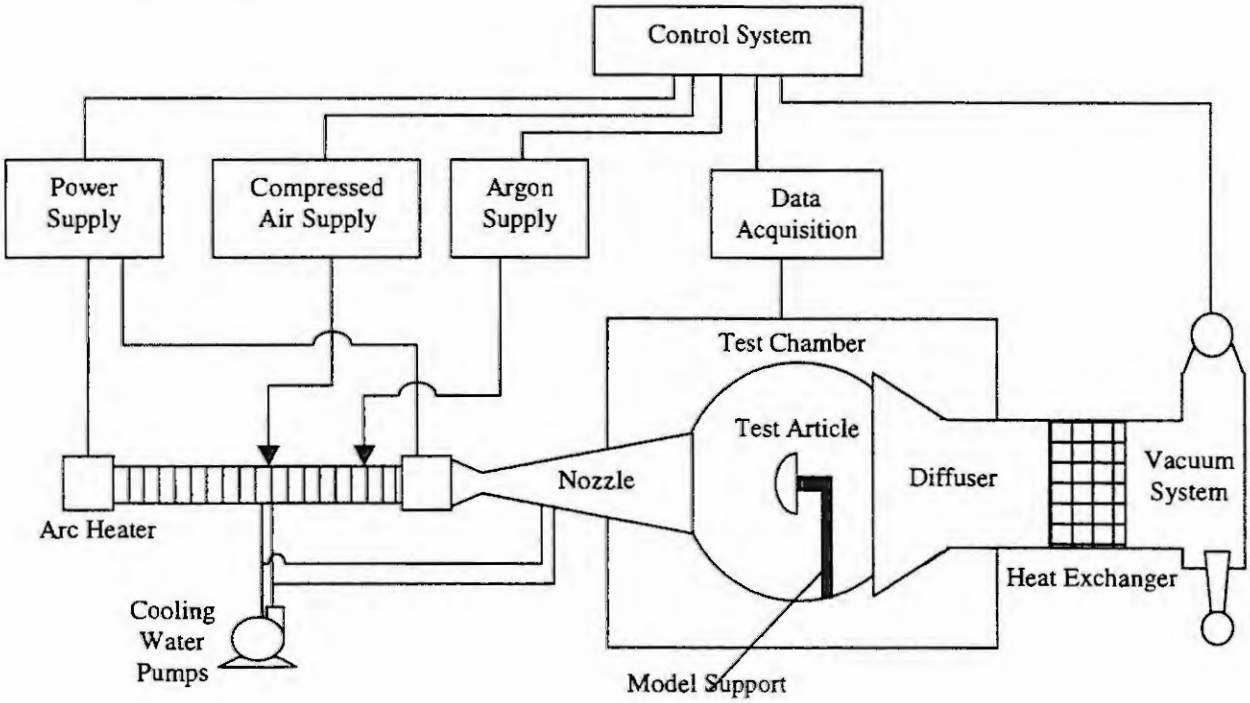


Figure 1: Schematic representation of *Scirocco*

Today the PWT DSS is going to be used not only as an aid to the design of the plant and of the control laws, but also as a primary tool for the plant operating. Indeed by means of a number of graphical interfaces, the test engineer will be supported in the choice of the right plant configuration, in the of design the appropriate control laws, and in the individuation of potentially critical operating conditions for each test to be performed.

The Software Organization

The PWT DSS has been implemented in Matlab/Simulink™ and is composed of two main simulation modules, two main graphical interfaces and a control law design environment. This SW is strictly connected to the Test Control Module entrusted with the *on-line* generation of the control commands for the execution of the desired tests. As for the simulation modules the *Detailed Dynamic Model* (DDM) allows to simulate, separately, one subsystem of the plant at a time in order to reproduce the time histories of the local variables of interest. Particular attention has been devoted to the detailed simulation models of those subsystems which play an important role in determining the performance of the whole facility, i.e. the power supply, the compressed air system, the vacuum system. The modelling of these subsystems is detailed to the level of the local regulation loops so that it will be also possible to evaluate the performance of such local control systems in the presence of the real load conditions. Moreover the DDM will be used to simulate the whole facility together with the test control algorithms in order to accurately evaluate the closed loop performance in terms of the time behaviors of all the variables of interest. Because of the high computational burden required by the DDM components, it will be mainly used to simulate the facility on limited time intervals of a test run, for example the ramp-up and ramp-down of current and air mass flow-rate, ramp variations of pressure and temperature on limited time intervals, etc.

The *Simplified Dynamic Model* (SDM) allowed *during the design phase* to design the test control algorithms, to verify the feasibility of the control algorithms on the computer hardware architecture, to evaluate the impact of the variation of some subsystem parameters on the PWT performance.

During the testing phase it will be used to check the correctness of the test control algorithms coding by running the real control system on the numerically simulated plant (real time simulation with hardware in the

loop), and to verify the feasibility of the facility acceptance tests both with numerically simulated plant and control system and with numerically simulated plant and real control system.

Finally *during the operating life of the facility* it will be used to compute the pre-programmed input commands needed to run both static and dynamic tests specified in terms of desired aerothermodynamic conditions on the test article; to tune the control system parameters for the execution of both static and dynamic tests in presence of test articles with thermal/geometrical characteristics different than those considered in the system design phase.

The Simulation Modules

From a system point of view the PWT can be schematized as a set of subsystems variously interconnected. A *Simulink block library* (see Figure 2) containing the models of the relevant subsystems was created. For the subsystems to be used both in the SDM and in the DDM modules, models with two different levels of detail are available. Also inverse models for the computation of the nominal *feedforward* control laws were produced. For a discussion on the mathematical and physical derivation of the models see [1], [2], [11], while for a discussion of the dynamic inversion of the models, the interested reader may refer to [9] and [10].

Test Article. The model of the test article is based on the numerical solution of the Heat Equations in a solid body [7]. Radiative and convective heat exchange phenomena are also taken into account. Due to the axial-symmetry of the test article, a two-dimensional and a one-dimensional model are available.

Arc Aerothermodynamics. The physical process governing the arc heater behavior is very complex. Moreover the *Scirocco* arc heater is the longest constricted-segmented arc heater ever built and can be considered a prototype. For this reason a numerical model based on computational aerothermodynamics is not reliable. Hence a low order model based on the fusion of theoretical predictions with experimental data collected on smaller arcs has been built (see [2]).

Arc Electric Load. With this block the nonlinear electric load offered by the arc heater to the power supply is modelled. Non-linear smoothing reactors, windings, cables, etc., concur to this load. (see [1]).

Divergent Nozzle. The divergent nozzle is another core component of *Scirocco*. Due to the hypersonic flow field, the phenomenon can be considered instantaneous if compared with the dominant dynamics of the test article. A neural network based model has been implemented on the basis of the results carried out with a code developed at CIRA which numerically solves the *Navier Stokes equations* in the presence of chemical non-equilibrium (see [3], [5], [6]).

Power Supply System. As for the simplified model, the PSS is considered as a whole with its local regulation loop and it is modelled with a first order linear time invariant system. On the contrary the detailed model takes into account the real waveforms generated by the thyristor controlled bridge. This level of detail requires very small integration steps and can be used only for simulations over a limited time interval.

Air Compressed Supply System. The air compressed system can be considered as a pneumatic network consisting of pipes, vessels, regulating valves, venturi, etc. In the simplified model it is modelled as a whole with its local regulation system as first order linear time-varying system while The detailed model is a high order nonlinear model simulating the mass flow-rates and pressure in a number of points of the pneumatic network.

Arc Air Injection System. The arc air injection system is the complex of pipelines, vessels, orifices and other pneumatic components feeding the arc chamber. It is modelled using the same approach used for the air compressed supply system. In the SDM it is considered together with the air supply system.

Vacuum System. While in the simplified simulation of the plant this subsystem can be neglected if we assume its correct working which assure a desired pressure in the test chamber, a detailed model is needed to verify the feasibility of certain trajectories and to compute the vacuum systems control commands. The model of this subsystems was carried out on the basis of the characteristic curves and times of the steam ejectors provided by the designer.

Heat Exchanger and Diffuser. These two subsystems composing the energy recovery system are modelled by means of semi-empirical formulas carried out by the project engineers combining numerical simulation results with experimental data.

Control System. The simulation model of the Control System is a non real-time version of the control SW implemented on the *Supervisor* to compute the set points for the power supply, air supply and vacuum system local regulation loops.

Other Subsystems. Other minor subsystems of the plant are also modelled to have information on their working conditions during the tests. However they do not have a significant impact on the variables of our interest. This is the case of the *Argon System* which guarantees the correct arc ignition and the arc heater electrodes protection but does not directly influence the plant variables of main interest.

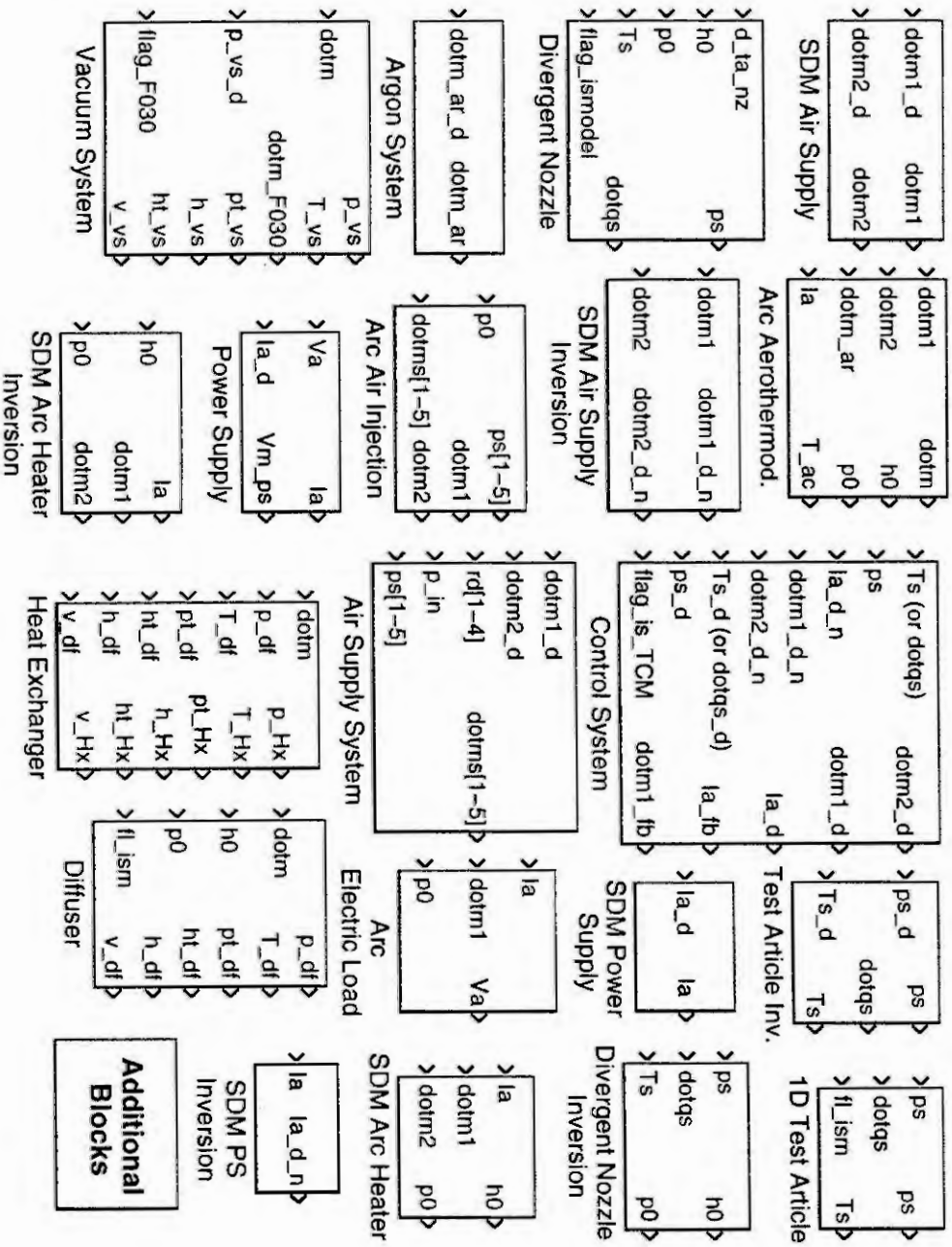


Figure 2: The main components of the PWT Simulation Block Library Components

The Test Control Module and the User's Interface

The PWT experimental test conditions are specified in terms of pressure and temperature time behaviors on a selected area of a specified test article. Two types of test articles representative of the space vehicle thermal protection systems are of interest: the *nose* and the *leading edge*. For each of them, two different types of tests will be performed: in the *static tests* the pressure and temperature at the stagnation point of the test article have to be kept constant for a given time interval in the pressure range [5, 175] *mbar* and in the temperature range [1273,

1973] *K*. In the *dynamic tests* the controlled variables have to follow time trajectories with time derivatives ranging in the intervals ±10 *K/s* and ± 1 *bar/s*.

Since, for each experimental test, the desired pressure and temperature of the test article are known in advance, the control commands are determined as the sum of two components: a *prepogrammed input command*, determined on the basis of the desired reference trajectories using the nonlinear simplified model of the plant; a *feedback command*, determined on-line by an outer feedback controller on the basis of the measured values of the controlled variables. This additional term is needed to compensate the errors in the preprogrammed input commands due to the unavoidable uncertainties and parametric variations in the model of the plant.

For the calculation of the feedforward control action inverse dynamic models of all the relevant subsystems of the plant have been built. Moreover two graphical interfaces allow the user to compute the nominal control action for the arc heater supply systems and the vacuum system. These interfaces also allow for the choice of the proper facility configuration and the detection of possible critical operating conditions.

As for the feedback control two standard gain scheduled SISO PID controller will be used: one for the arc current-stagnation temperature and the other for the arc mass flow-rate-stagnation pressure input-output channel respectively.

In order to tune the parameters of the PIDs controller (see [4]) a Parameter Tuning SW Library was implemented. A family of linear plants is firstly derived by linearising the simplified nonlinear model of the plant at a specified number of equilibrium conditions throughout the operating envelope. To simplify both the design and analysis of the controller, for each linearized model a reduced order model is derived. Each operating condition is defined by the nominal arc current and mass flow-rate so obtaining a discrete point state space description of a Linear Parameter Varying system. For each frozen parameter value two PID controller are designed. Finally a gain scheduling scheme consisting of switching the linear controllers along the system trajectory is adjusted. As pointed out in [12], when using these fixed parameter designs, the performance or even stability in the presence of parameter variations are not assured. Extensive campaign of simulations with the nonlinear plant are, therefore, required to verify stability and performance of the closed loop system.

Conclusions

In this paper a software environment for the simulation and the control law design of the *Scirocco* Plasma Wind Tunnel was presented. Two main modules have been built for the plant dynamic simulation: a simplified simulator basically oriented to the control law design and a detailed simulator. A complete library of the subsystem models has been built together with a number of graphical operator interface panels. A preliminary structure for the plant test control module was also described. Future work will be devoted to the calibration of the proposed dynamic simulation models and of the proposed control laws on the real plant.

References

1. Amato, F., Mattei, M. and Pironti, A., Robust Control of a Power Supply System for an Arc Heater. *Proceedings of the 5th IEEE Conference on Control Application*, Detroit (MI), 1996.
2. Ambrosino G., Celentano G., and Mattei M., A Control Design Oriented Mathematical Model for the *Scirocco* Plasma Wind Tunnel, submitted to *Mathematical and Computer Modelling of Dynamical Systems*.
3. Anderson, J.D., Hypersonic and High Temperature gas Dynamics. McGraw Hill, New York, 1989.
4. Astrom, K.J., and Hagglund, T., PID Control-Theory, Design and Tuning, 2nd ed.. Instrument Society of America, Research Triangle Park, NC, 1995.
5. Borrelli, S., Mattei, M., and Schettino, A. A simplified approach to the simulation of the aerodynamics in *Scirocco* plasma wind tunnel. *Proceedings of the Second European Symposium on Aerothermodynamics for Space Vehicles*, Nordwijk (ND), 1994.
6. Hirsch, C., Numerical computation of internal and external flows. John Wiley and Sons, Brussels, 1987.
7. Kreith, F., Principles of Heat Transfer. Educational Publisher, New York, 1973.
8. Ljung. L, System Identification, Theory for the User, Prentice Hall, Englewood Cliffs, NJ, 1987.
9. Mattei M., Modelling and Control of a Modern Plasma Wind Tunnel. PhD Thesis (*in italian*), Dipartimento di Informatica e Sistemistica, Università di Napoli Federico II, Napoli, 1997.
10. Mattei M., A Robust Trajectory Following Control System for Space Vehicle Testing, Proceedings of the *IEEE 37th Conference on Decision and Control*, Tampa (FL) 1998.
11. Mattei M., A Robust Multiple PI Controller for the Air Distribution into an Arc Heater, *Proceedings of the '99 European Control Conference*, Karlsruhe, Germany, 1999.
12. Shamma, J.S., and Athans, M., Gain Scheduling: Potential Hazards and Possible Remedies. *IEEE Control System Magazine*, Vol. 12, 1992.

THE SYSTEM OF PANTOGRAPH AND CATENARY: MATHEMATICAL MODELS AND NUMERICAL TECHNIQUES

M. Herth, B. Simeon
Center for Scientific Computing and Mathematical Modelling
University of Karlsruhe

Abstract. In this paper we discuss a mathematical model and its numerical simulation for the interaction of pantograph and catenary. First the benchmark problem of Arnold/Simeon [1] for the catenary is extended and a more detailed pantograph model is used. Afterwards the semi discretization by the finite element method and the time integration are described. In this context numerical techniques like GGL-stabilization and superconvergent patch recovery are applied. The latter yields an error estimation for the finite element grid and shows the critical points of the system.

Introduction

Pantograph and catenary dynamics is the most critical part in the energy transmission of high-speed trains. Often, oscillations in the contact wire occur such that the contact force between pantograph and catenary varies strongly and the contact may even get lost. Therefore, constructive changes for both pantograph and catenary are under development. Design criteria include the permanent contact of pantograph head and contact wire at high speed and the reduction of both aeroacoustic noise and wear [8]. Dynamical simulation plays an important role in this development since prototypes and measurements are very expensive.

In the present paper we investigate mathematical models and numerical simulation techniques for this coupled system. Our approach is based on a descriptor formulation where the contact condition is treated as a unilateral constraint and where a continuous model for the catenary is combined with a mechanical multibody system for the pantograph. This results in partial differential equations (PDE's, catenary) and differential-algebraic equations (DAE's, pantograph). The simulation results show that contact losses appear if the velocity exceeds 200 km/h.

The paper is organized as follows: First, we introduce shortly the coupled system for the equations of motion and analyse its structure. Next, space and time discretization techniques are discussed. In particular, the quality of the FE grid is assessed in terms of the superconvergent patch recovery of Zienckiewicz/Taylor [11]. Finally, several simulation runs at different speeds demonstrate the dynamic behavior.

Mathematical models

In this section we introduce the mathematical models for pantograph and catenary. The pantograph model contains seven degrees of freedom and five masses, see Fig. 1, and describes, in addition to the vertical motion, the rotation of the pan head (q_3) and two contact points, see [7] for more details. The latter is necessary to take the zigzag-course of the contact wire into account.

The model for the catenary consists of carrier, contact wire and 14 droppers, see Fig. 2. Furthermore, we model the registration arm at the midpoint as a spring damper system, in contrast to those at the boundary, which are fixed.

The carrier is modelled as a string, but we use an Euler-Bernoulli-beam for the contact wire to get a continuous first derivative at the contact point. The equations of motion contain the density ρ, cross section area A, damping constant β, normal force T and the bending stiffness EI in case of the beam. Moreover, w_c and w_w denote the displacement of the carrier resp. contact wire.

$$\rho_c A_c \ddot{w}_c + \beta_c \dot{w}_c = T_c w_c'' - \rho_c A_c g - \sum_{j=1}^{n_d} F_{dc,j} \tag{1}$$

$$\rho_w A_w \ddot{w}_w + \beta_w \dot{w}_w = -E_w I_w w_w'''' + T_w w_w'' - \rho_w A_w g - \sum_{j=1}^{n_d} F_{dw,j} + \sum_{j=1}^{n_s} F_{pw,j} + F_r. \tag{2}$$

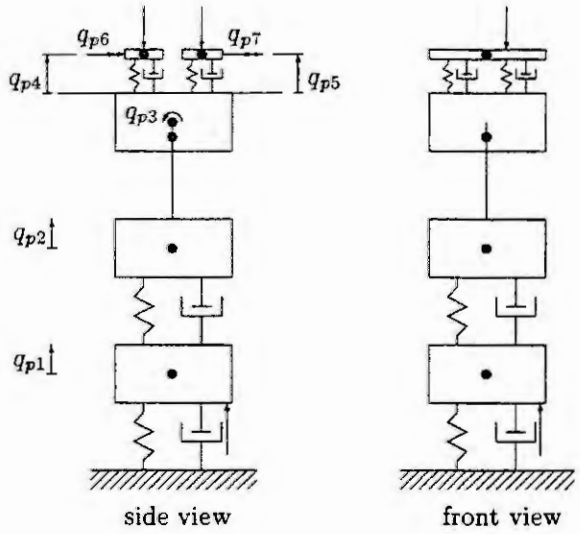


Figure 1: Pantograph model with five masses and seven degrees of freedom

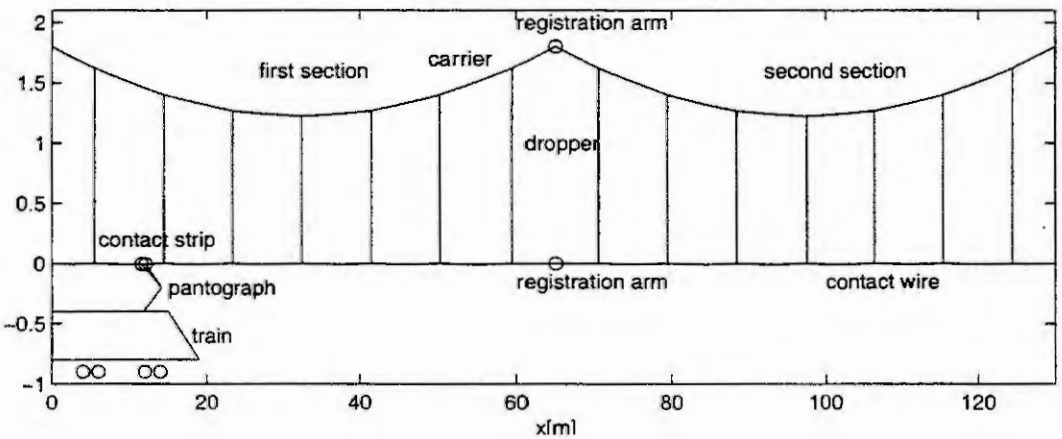


Figure 2: System of catenary and pantograph

The constraint forces $F_{dc,j}$, $F_{dw,j}$, $F_{pw,j}$ and F_r are point forces and therefore contain a delta function. The dropper forces $F_{dc,j}$ and $F_{dw,j}$ include also the inertia terms of droppers and Lagrange multipliers $\lambda_{d,j}$, which combine these two equations. The equation for the pantograph model

$$M_p\ddot{q}_p = -D_p\dot{q}_p - C_pq_p - B^T\lambda_p + F_p \tag{3}$$

is also connected to (2) by Lagrange multipliers $\lambda_{p,j}$. They represent the contact forces between contact wire and pantograph. These couplings lead to the following constraint conditions

$$0 \leq w_w(x_{d,j},t) - w_c(x_{d,j},t) + l_{d,j}, \quad j = 1,\ldots,n_d \tag{4}$$

$$0 \leq w_w(x_{p,j}(t),t) - b_{p,j}q_p, \quad j = 1,\ldots,n_s, \tag{5}$$

which are unilateral since the droppers have the possibility to slacken and the pantograph can loose the contact to the catenary.

The equations (1) to (5) include two partial differential equations, one ordinary differential equation and two algebraic equations. Hence they form a partial differential algebraic equation (PDAE).

Semi-discretization

To solve the above system we use the method of lines and start with the semi-discretization by finite elements. For this purpose, we multiply the equations of motion (1) to (5) with a testfunction v_w resp. v_c and integrate over x. This leads to the weak form of the equations and by projection onto a finite dimensional subspace, e. g. a finite element space, we obtain the semi discretization in terms of the differential algebraic equation (DAE)

$$\begin{aligned} M_c\ddot{q}_c + D_c\dot{q}_c &= -S_c q_c + b_c - H_c^T\lambda_d \\ M_w\ddot{q}_w + D_w\dot{q}_w &= -(S_w + K_w)q_w + b_w + H_w^T\lambda_d + F_w^T(t)\lambda_p \\ M_p\ddot{q}_p &= -D_p\dot{q}_p - C_p q_p - B^T(q_p)\lambda_p + F_p \\ 0 &\leq H_w q_w - H_c q_c + l_d \\ 0 &\leq F_w(t)q_w - B(q_p). \end{aligned} \tag{6}$$

Details can be found in [1]. We introduce the mass matrix M, damping matrix D and stiffness matrix S and combine the applied forces to the vector b, while the indices w, c and p denote contact wire, carrier and pantograph. These equations can again be summarized to the more convenient, linear time-variant form

$$\begin{aligned} M\ddot{q} + D\dot{q} &= -Sq + b + G(t)^T\lambda \\ 0 &= G(t)q + z \end{aligned} \tag{7}$$

with the vectors $q = (q_w, q_c, q_p)^T$, $b = (b_w, b_c, F)^T$, $z = (l_d, 0)^T$ and $\lambda = (\lambda_d, \lambda_p)^T$. Here, the unilateral constraints are assumed to be active.

Time integration

The resulting DAE (7) has the differential index three. This property leads to order reduction and so we have to use a stabilized formulation. We apply the GGL-stabilization of Gear, Gupta, Leimkuhler, see [3, 5], which uses the velocity constraint condition and adds an Lagrange multiplier μ to the system. Then we can append the displacement constraint condition as an invariant.

$$M\dot{q} = Mv + G^T(t)\mu \tag{8}$$

$$M\dot{v} = -Dv - Sq + b + G^T(t)\lambda \tag{9}$$

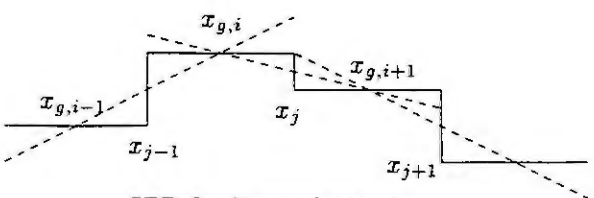
$$0 = \dot{G}(t)q + G(t)v. \tag{10}$$

$$0 = G(t)q + z \tag{11}$$

The system (8) to (11) can be integrated by implicit methods like BDF2. The unilateral constraints are assumed to be active at the beginning. If a constraint force λ reaches zero, the corresponding constraint becomes inactive and the time step is repeated. The calculation is continued until the constraint becomes active again.

Error estimation with superconvergent patch recovery

To examine the accuracy of the computation, we apply the superconvergent patch recovery (SPR), which yields an error estimation for the stresses. It is based on the superconvergence of gauss points, see [11] for the theoretical background.



SPR for linear finite elements

SPR calculates an approximation polynom through the gauss points of two neighbouring elements and computes the difference between this polynom and the finite element solution. The polynom

$$u'(x) = P(x)d = [1, x, \ldots, x^m] \cdot [d_1, d_2, \ldots, d_{m+1}]^T$$

should therefore minimize the potential

$$\Pi = \sum_{i=1}^{2m} (w'(x_{g,i}, t_k) - P(x_{g,i})d)^2$$

with $m = 1$ for linear elements and $m = 3$ for kubic elements. Hence, the linear polynom is exact but the kubic one is a least mean square approximation. The differentiation of Π with respect to d leads to the linear equation $Ad = b$ with

$$A = \sum_{i=1}^{2m} P(x_{g,i})^T P(x_{g,i}) \quad \text{and} \quad b = \sum_{i=1}^{2m} P(x_{g,i})^T w'(x_{g,i}, t_k),$$

which can be solved numerically, see [6]. To avoid approximation polynoms through elements next to a discontinuity, we do not use SPR at critical points like droppers and registration arm. Instead we calculate the middle value of the neighbouring approximation polynoms at these points.

Simulation results

The numerical solution is shown in Fig. 3 and Fig. 4. Fig. 3 displays the motion of the pantograph at a velocity $v_0 = 48\,\mathrm{m/s}$. On the left picture one can see the displacement of the two contact strips $q_{p4,5}$

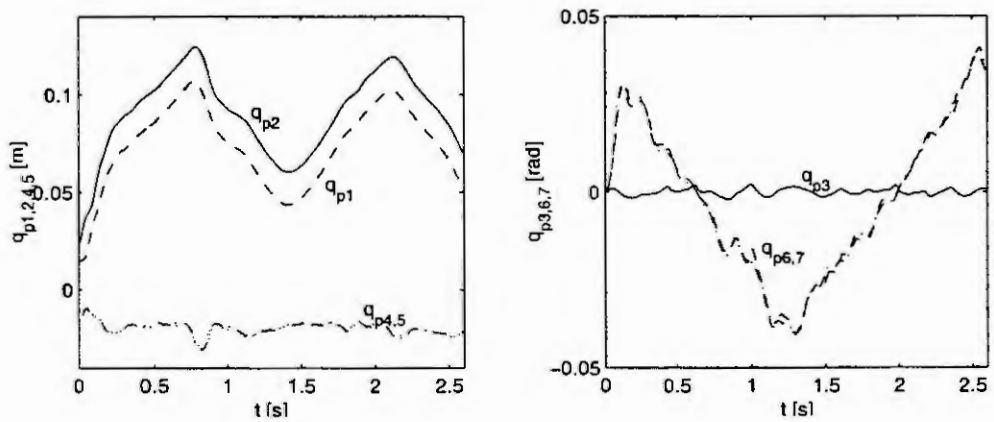


Figure 3: Translatory and rotatory degrees of freedom of the pantograph for $v_0 = 48\,\mathrm{m/s}$

and the two masses at the lower part of the pantograph q_{p1} and q_{p2}. The behaviour of the contact strips is very similar, their displacement is relatively small and without strong oscillations.

The motion of the other two masses is comparable to the motion of a beam under a constant moving force, see [4]. This shows that the influence of the droppers is negligible for the overall movement of the pantograph. The right picture shows the rotation of the pan head q_{p3} and the contact strips $q_{p6,7}$. Due to small motion of the pan head, the linearization of the equations of motion for the pantograph is sufficient. The rotation of the two contact strips is again very similar and follows the zigzag-course of the contact wire.

Fig. 4 shows the contact forces for the velocities $v_1 = 48\,m/s$, $v_2 = 55\,m/s$ and $v_3 = 63\,m/s$. They are not smooth but oscillate which leads to fading of the contact strips. No slackening of droppers or contact losses occur for v_1, but at speed v_2 contact losses of the last strip appear.

At the highest velocity v_3 we obtain both slackening of droppers and contact losses of the pantograph. The frequency of occurence is summarized in Table 1.

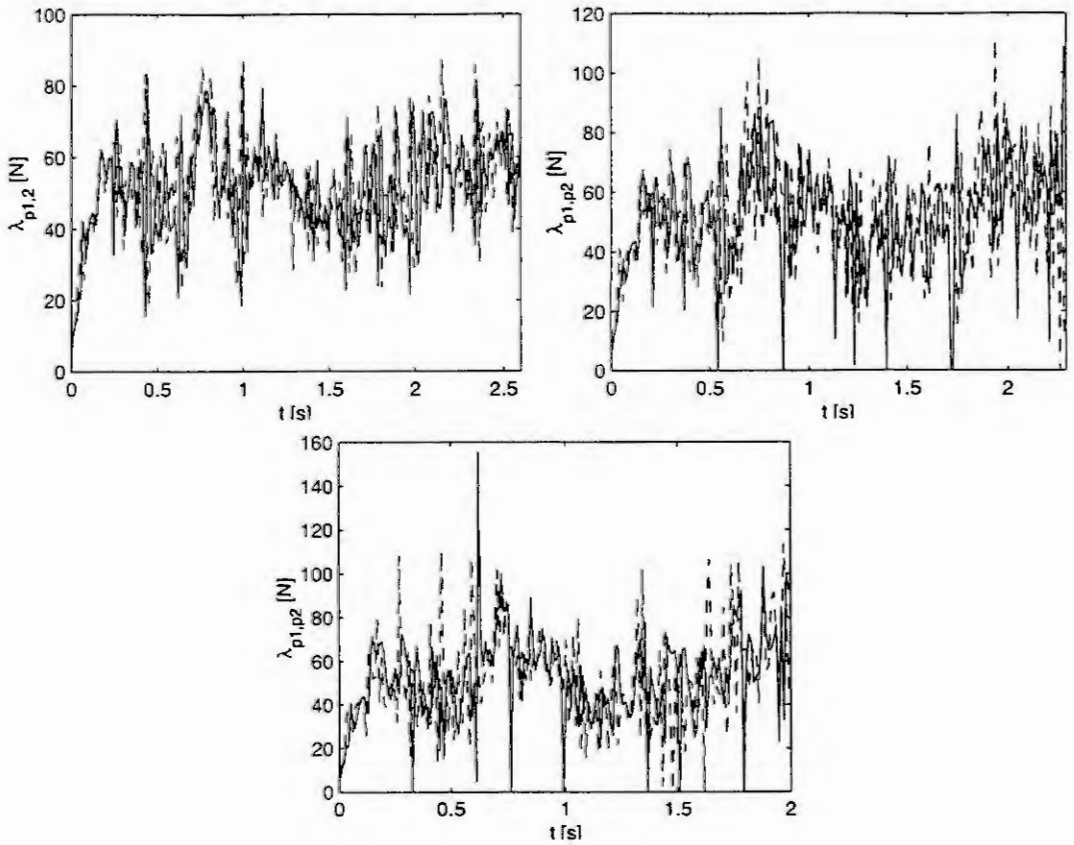


Figure 4: Contact forces for $v_1 = 48\,m/s$, $v_2 = 55\,m/s$ and $v_3 = 63\,m/s$

v	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_9	d_{10}	d_{11}	d_{12}	d_{13}	d_{14}	s_1	s_2
55	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	3
63	0	1	0	1	0	1	2	0	0	1	0	1	1	2	9	4

Table 1: Frequency of occurence of inactive constraints for the droppers d_1 to d_{14} and the contact strips s_1 and s_2

Remarkably the last strip looses the contact more often than the first because of the vibrations generated by the first. Furthermore the droppers directly before the registration arms in the middle and at the right side d_7 and d_{14} slacken both twice. In this situation the pantograph is at its highest position and is pressed down to the registration arm. This leads to a relatively high contact force as one can see in Fig. 4.

Table 1 includes also three contact losses at velocity $v_2 = 55\,\mathrm{m/s}$ for the contact strip s_2. These appear all during the last milliseconds when the pantograph is at the end of the catenary model and are therefore not relevant.

Fig. 5 displays the SPR error estimation for the contact wire. The left picture indicates that the error at the droppers is much higher than for the areas in between. At a later date the error in the contact wire between droppers increases but the error is still highest at the droppers. This shows that one can improve the calculation, if a non equidistant grid is used, which is done in [6] for a smaller model.

Conclusions

Effects like contact losses and slackening of droppers can only be resolved by numerical methods that take the local character of the interaction between pantograph and catenary into account. The mathematical model introduced here allows the application of such methods and offers much flexibility, in contrast to

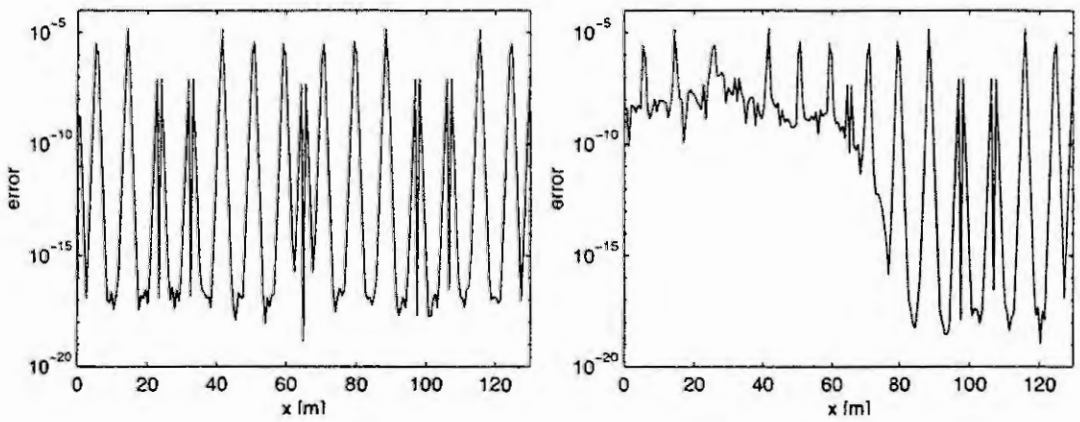


Figure 5: SPR error estimation for the initial value (left) and $t = 0.472\,\mathrm{s}$ (right) at $v_1 = 48\,\mathrm{m/s}$

approaches that intertwine modelling and simulation. Moreover, the SPR error indicator shows critical parts of the proposed FE discretization and could be a good basis for adaptive strategies.

References

[1] M. Arnold, B. Simeon. Pantograph and catenary dynamics: a benchmark problem and its numerical solution. To appear in Appl. Numer. Math.

[2] R. Courant, D. Hilbert. Methoden der Mathematischen Physik I. Springer-Verlag, Heidelberg New York, 1968.

[3] C.W Gear, G.K. Gupta, B.J. Leimkuhler. Automatic integration of the Euler-Lagrange equations with constraints. J. Comp. Appl. Math. 12&13, 77-90, 1985.

[4] P. Hagedorn. Technische Schwingungslehre. II. Lineare Schwingungen kontinuierlicher mechanischer Systeme. Springer-Verlag, Berlin Heidelberg, 1989.

[5] E. Hairer, G. Wanner. Solving Ordinary Differential Equations. II. Stiff and Differential-Algebraic Problems. Springer-Verlag, Berlin Heidelberg New York, 1996.

[6] M. Herth. Simulation der Dynamik von Fahrleitung und Stromabnehmer. Diploma thesis, 1999.

[7] K. Petri. Vergleichende Untersuchung von Berechnungsmodellen zur Simulation der Dynamik von Fahrleitung-Stromabnehmer-Systemen. Dissertation, Heinz-Nixdorf-Institut, Universität-GH Paderborn, Germany, 1996.

[8] G. Poetsch, J. Evans, R. Meisinger, W. Kortüm, W. Baldauf, A. Veitl, J. Wallaschek. Pantograph/catenary dynamics and control. Vehicle System Dynamics, 28:159-195, 1997.

[9] G. Poetsch, J. Wallaschek. Simulating the dynamic behaviour of electrical lines for high-speed trains on parallel computers. In Proceedings of the International Symposium on Cable Dynamics, S. 565-572, A. I. M. Liège, 1995.

[10] B. Simeon, M. Arnold. Coupling DAE's and PDE's for simulating the interaction of pantograph and catenary. To appear in Math. and Computer Model. of Systems.

[11] O. C. Zienckiewicz, R. L. Taylor. The Finite Element Method. Mc Graw Hill, London, 1994.

Natural Coordinates and Mechanical DAE

C. Kraus, M. Winckler [1]
Interdisciplinary Center for Scientific Computing (IWR), University of Heidelberg
INF 368, D-69120 Heidelberg
Christian.Kraus@iwr.uni-heidelberg.de
Michael.Winckler@iwr.uni-heidelberg.de

Abstract We will present a modified version of natural coordinates introduced by García de Jalón and Bayo et al. [1]. Natural coordinates are suitable for easy and flexible modeling of multi body systems. Our version introduces no redundant constraints. That allows for a direct solution of the linear system of the full descriptor form. Although one will work with many dynamic variables, the resulting system is very cheap to generate and sparse. The sparse structure can be exploited by a block oriented rational cholesky algorithm, to get linear complexity with respect to the number of bodies and constant overhead for kinematic loops.

Coordinates in Multi–Body Systems

When modeling the dynamics of mechanical systems one of the key aspects is the choice of coordinates. Several approach to this issue exist and the question of an optimal choice is an area of ongoing research. Tree-structured mechanisms without closed kinematic loops can be model recursively traversing trough the the kinematic chains. This usually leads to a set of ordinary differential equations (ODEs) for the equations of motion.

Alternatively a variety of *redundant coordinate systems* for modeling multi-body systems (MBS) on the basis of Lagrangian equations of the first kind are in use. This leads to the well-known set of differential algebraic equations of index 3 for MBS. Index-reduction for numerical treatment of this DAE leads us to the *full descriptor form* of index 1:

$$\begin{aligned} \dot{p} &= v \\ \dot{v} &= a \\ \begin{pmatrix} M & G^T \\ G & 0 \end{pmatrix} \begin{pmatrix} a \\ \lambda \end{pmatrix} &= \begin{pmatrix} f \\ \gamma \end{pmatrix} \\ g_p(p) &= 0 \\ g_v(p) &= 0 \end{aligned} \tag{1}$$

with positions p, velocity $v = \frac{dp}{dt}$, acceleration $a = \frac{dv}{dt}$, mass matrix M, forces f, kinematic constraints g_p, velocity constraints g_v, acceleration independant part of the acceleration constraints γ, Lagrange - multiplier λ and constraint matrix $G = \frac{dg_p}{dp}$.

A unifying idea for most modelling techniques is to establish a local coordinate system in each body of the MBS. The coordinate choice is usually the choice of an efficient parameterization of the affine transformation between the global (world) and each of the local (body) coordinate systems. Reference point coordinates e.g. model the kartesian coordinates of the origin of the local coordinate system together with three angles (Euler angles). While this leads to the most compact possible set of coordinates, the drawback of singular positions is the cause of a variety of numerical problems. Overcoming this drawback is only possible with redundant coordinate systems, since any parameterization of a free body in 3D with exactly 6 coordinates has at least two such singular points.

Our modeling method of choice uses *natural coordinates* as described by García de Jalón and Bayo in [1]. Although this method uses the huge number of 12 coordinates per body, we will show, that the resulting equations are very simple und have a nice structure to exploit. This leads directly to an algorithm of linear complexity, that is competitive, if all substructures are treated efficiently.

General Affine Transformation leads to Natural Coordinates

Natural coordinates parameterize the underlying affine transformation in a natural and more general way. In most other methods the affine transformation is specialized to the combination of a translational part of the origin and

[1]This work is funded by the German Department of Science and Technology within the research grant OPTIMIERUNG VON MOTORKOMPONENTEN and conducted under the advice of Prof. Dr. H.G. Bock and Dr. J. Schlöder.

a *rotational* part. But the orientation matrix need not be an element of $\mathcal{SO}(3)$. A more general view, that allows scaling and shearing can be used to adjust the coordinate system to the local topology of the model.

An affine transformation of a body–fixed point r expressed in local coordinates to his global representation $\bar{r}$ is given by

$$\bar{r} = o + Xr \tag{2}$$

where $o \in \mathcal{R}^3$ is the origin of the local coordinate system and $X \in \mathcal{R}^{3\times 3}$ is the corresponding orientation matrix.

Our method does *not* parameterize X, but uses *all* 9 entries of X as dynamic variables. Because of this relation between the dynamic variables and the orientation matrix every local point or vector can be transformed to its global representation by a *linear* equation of the dynamic variables.

$$\begin{aligned}
\bar{r} &= o + Xr = o + (e_1|e_2|e_3)r \\
&= 1o + r_1 e_1 + r_2 e_2 + r_3 e_3 \\
&= \underbrace{\begin{pmatrix} 1 & & & r_1 & & & r_2 & & & r_3 & & \\ & 1 & & & r_1 & & & r_2 & & & r_3 & \\ & & 1 & & & r_1 & & & r_2 & & & r_3 \end{pmatrix}}_{C_r} \underbrace{\begin{pmatrix} o \\ e_1 \\ e_2 \\ e_3 \end{pmatrix}}_{p} \\
&= C_r p
\end{aligned} \tag{3}$$

where $C_r \in \mathcal{R}^{3\times 12}$ is defined by the local affine coefficients and $p \in \mathcal{R}^{12}$ are the dynamic variables of each body. The equation holds true for the first and second derivative with the same matrix C_r.

Remark Besides affine points the modeling process uses *translation* vectors, which are translational invariant. Setting the leading 3x3 block of C_r to zero gives the correct transformation, while preserving the structure.

Remark While the basic technique uses one point and three vectors, variants with different configurations are more suitable to model special local topologies, e.g. for a body with two spherical joints attached one might use two points (located in the joints) and two vectors. However, the resulting modifications keep the above structure of C_r.

Deriving the mass matrix

The mass matrix will be derived from the general form of the kinetic energy of an system with generalized coordinates q

$$T = \frac{1}{2}\dot{q}^T M \dot{q} = \frac{1}{2}\int_V \rho\, \dot{q}^T \dot{q}\, dV \tag{4}$$

Making use of the first derivative of transformation (3) we can write the kinetic energy with respect to the dynamic variables p

$$T = \frac{1}{2}\int_V \rho\, \dot{p}^T C_r^T C_r \dot{p}\, dV \tag{5}$$

Because the integral is independent of the dynamical variables, which are derived from the transformation of the whole *rigid* body, they could be moved out of the integral.

$$T = \frac{1}{2}\,\dot{p}^T \int_V \rho\, C_r^T C_r\, dV\, \dot{p} = \frac{1}{2}\,\dot{p}^T M \dot{p} \tag{6}$$

We get an bi–linear form with respect to the velocity, so we can establish the mass matrix for natural coordinates. The mass matrix is constant (as in many formalisms with an body–fixed coordinate systems) and sparse, a fact that follows directly from the structure of the transformation matrix C_r. It can be computed directly or by using the center of mass and the matrix of inertia computed with respect to Eulerian angles.

Modeling rigid bodies

In natural coordinates the orientation matrix X must not be an element of $\mathcal{SO}(3)$. However, when modeling *rigid bodies* a set of algebraic relations between the natural coordinates exist. These so called rigid body constraints restrict the coordinates to the suitable 6–dimensional subspace. These relations keep the length of the axis and the angles between them fixed:

$$\begin{array}{lcll} g^{RC}_{p,i} & = & \frac{1}{2}e_i^T e_i - l_i^2 & (i = 1 \ldots 3) \\ g^{RC}_{p,4} & = & e_1^T e_2 - a_1 & \\ g^{RC}_{p,5} & = & e_1^T e_3 - a_2 & \\ g^{RC}_{p,6} & = & e_2^T e_3 - a_3 & \end{array}$$

All these equations are polynomial of *second order*, so they and their derivatives, which are used in the full descriptor from, are very cheap to compute. The resulting constraint matrix is *linear* with respect to the dynamic variables and has a special structure.

$$G^{RC}(p) = \frac{\partial g_p^{RC}}{\partial p}(p) = \begin{pmatrix} e_1^T & & \\ & e_2^T & \\ & & e_3^T \\ e_2^T & e_1^T & \\ e_3^T & & e_1^T \\ & e_3^T & e_2^T \end{pmatrix} \tag{7}$$

The computation of γ can be rather time consuming in other formulations, because it corresponds to the second derivative of a possibly highly nonlinear kinematic constraint. In numerical experiments the computation of γ consumed up to 80% of the time to evaluate the model. In our formulation the computation of γ is as cheap as the evaluation of the kinematic constraints.

$$\begin{array}{lcll} \gamma_i^{RC} & = & -2\dot{e}_i^T \dot{e}_i & (i = \{1,2,3\}) \\ \gamma_i^{RC} & = & -2\dot{e}_j^T \dot{e}_k & (i = \{4,5,6\}) \end{array} \tag{8}$$

Defining joints by linear independent constraints of second order

Introducing joints restricts the manifold of feasible points even further. So modeling joints can be achieved by imposing additional joint constraints. The two key concepts for modeling standard joints are restricting translational movement in one direction and rotational movement around an axis, both of which can be derived using the concept of parallelism.

A formulation introduced by García de Jalón and Bayo uses the vector product. It leads to a set of three *linearly dependent* equations. But for numerical treatment a constraint matrix with full row rank is advantages. An adopted formulation using two scalar products of the following form

$$a \| b \iff \exists\, h_1 \perp h_2, a \perp h_1, a \perp h_2 : \; b \perp h_1 \;\wedge\; b \perp h_2 \tag{9}$$

avoids this problem.

Example We want to derive the constraints of a revolute joint. We have to fix two bodies at a common point with local representations b_1, b_2. With transformation (3) we get

$$g_{p;1..3}(p_1, p_2) = b_1(p_1) - b_2(p_2) = C_{b1}p_1 - C_{b2}p_2 = 0 \tag{10}$$

Furthermore we have to forbid rotations other than around the rotation axis (local coordinates d_1, d_2).

$$g_{p;4}(p_1, p_2) = d_1(p_1)^T d_2(p_2) = (C_{d1}p_1)^T C_{h1}p_2 = 0 \tag{11}$$

$$g_{p;5}(p_1, p_2) = d_1(p_1)^T d_2(p_2) = (C_{d1}p_1)^T C_{h2}p_2 = 0 \tag{12}$$

where h_1 and h_2 are perpendicular to d_2.

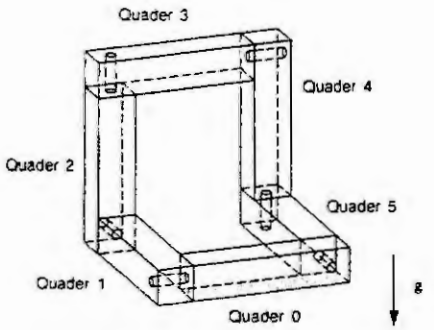
All other standard joint used in robotics (spherical, cylindrical, prismatic and universal) can be modeled in a similar low–order fashion.

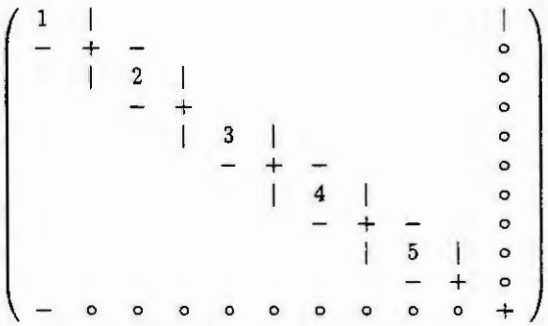
Exploiting the block–sparse structure leads to an $\mathcal{O}(n)$ algorithm

The local topology of bodies and joints leads to a block–sparse structure of the linear system (1). In the original formulation of García de Jalón and Bayo *point sharing*, that is sharing of axis and points between bodies, is used to shrink the system. This leads to a decomposition algorithm with quadratic complexity, which is useful for small models.

Instead of reducing the system via point sharing our approach keeps the full structured matrix. For tree– structured systems there exists a block–ordering that results in a block–banded matrix. Here the augmented system of the mass matrix and the rigid-body constraints corresponding to each body is placed on the main diagonal and the constraint matrix block corresponding to joints are placed on the first subdiagonal. This matrix can be decomposed in linear time using a block–oriented rational cholesky algorithm. If joints leading to closed kinematic loops are present, off-diagonal blocks are introduced. These propagates fill–in in the magnitude of the size of the loop.

Example The structure of the linear system is illustrated by a model of the 6–bar–mechanism. The model consists of 6 bodies (one fixed and not modeled) connected by 6 revolute joints in a special manner. The resulting system has one closed kinematic loop. It is well known, that this model has one degree of freedom, so one of the introduced constraints is linear depended to the others. This was detected by a rank analysis of the system and the corresponding constraint was eliminated beforehand.





Here the numbers represent the augmented systems of each body consisting of the mass matrix and the rigid body constraint matrix. The joint constraint matrix is represented by $-$ and $|$ for the transposed matrix. $+$ and $\circ$ are introduced by the decomposition algorithm, where $\circ$ is additional fill–in propagated by the closed kinematic loop.

Conclusion

Natural coordinates are a powerful tool for modeling mechanical systems. The representation of the local topology in the system matrix and a simple and flexible underlying transformation overcompensate for the apparent „drawback" of a redundant coordinate approach. First investigations by v. Schwerin about computation time show that natural coordinates can compete with other modeling techniques [3]. The $\mathcal{O}(n)$ - algorithm outlined in this article and a sparse substructure make it especially competitive for systems with a large number of bodies.

The software package MBSNAT (Multi Body System simulation with NATural coordinates) by Kraus supplies an object–oriented implementation of model generator and linear algebra for use in any suited MBS integration package (e.g. MBSSIM by v. Schwerin and Winckler).

References

[1] J. Garcìa de Jalòn and E. Bayo. *Kinematic and Dynamic Simulation of Multibody Systems*. Springer, 1994.

[2] G. Lester and W. Schiehlen. *Benchmark–Beispiele des DFG-Schwerpunktprogrammes "Dynamik von Mehrkörpersystemen"*. Zwischenbericht ZB-64. Institut B für Mechanik, Universität Stuttgart, 1991.

[3] R. v. Schwerin. *Numerical Methods, Algorithms and Software for Higher Index Nonlinear Differential-Algebraic Equations in Multibody System SIMulation*. PhD thesis, Universität Heidelberg, 1997.

Mathematical Problems in Circuit Simulation

C. Tischendorf and D. Estévez Schwarz [1]

Circuit simulation is a standard task for the computer-aided design of electronic circuits. The transient analysis is well understood and realized in powerful simulation packages for conventional circuits. But further developments in the production engineering lead to new classes of circuits which cause difficulties for the numerical integration. The dimension of circuit models can be quite large (10^5 equations). The complexity of the models demands a higher abstraction level. Parasitic effects become dominant. The signal to noise ratio becomes smaller. We want to focus in this paper on three essential problems from a mathematical point of view, the DAE-index, consistent initial values and asymptotic stability.

1 Structure of CAD-based systems for integrated circuits

The modified nodal analysis (MNA) is a widely used modeling technique which enables an automatic generation of the network equations under conservation of the circuit structure. At a first glance, it leads to differential algebraic equations (DAEs) of the form

$$C(x,t)\dot{x} + f(x,t) = 0, \tag{1}$$

where $C(x)$ is a singular matrix and x consists of all nodal voltages and the currents through current controlled elements. In case of the charge oriented MNA, the charges of capacitive elements as well as the fluxes of inductive elements are additionally included. A closer look onto the systems provides a special structure that can and should be exploited by numerical integrators.

The model description bases on five basic network elements. The static behavior is described by nonlinear controlled voltage sources, current sources and resistances. The dynamic behavior is reflected by constant or controlled capacitances and inductances. Splitting the incidence matrix A of the network elements into the element-related incidence matrices $A = (A_C, A_L, A_R, A_V, A_I)$, where A_C, A_L, A_R, A_V, and A_I describe the branch-current relations for capacitive branches, inductive branches, resistive branches, branches of voltage sources and branches of current sources, respectively, we obtain a system of the form

$$A_C \frac{dq(A_C^T e, t)}{dt} + A_R r(A_R^T e, t) + A_L j_L + A_V j_V + A_I i(A^T e, j_L, j_V, t) = 0, \tag{2}$$

$$\frac{d\phi(j_L, t)}{dt} - A_L^T e = 0, \tag{3}$$

$$A_V^T e - v(A^T e, j_L, j_V, t) = 0. \tag{4}$$

where e are the node potentials and $j_{L/V}$ are the current vectors of inductances/voltage sources. The functions q and ϕ describe the voltage-charge and current-flux relations for the dynamical elements. The controlling functions of current sources and voltage sources are represented by i and v.

2 DAE index of the network equations

Powerful numerical methods like the BDF (Backward Difference Formulae) method can be applied directly to DAEs of the form (1). They are often used successfully in

[1] Humboldt-University of Berlin, Germany

simulation packages. However they may fail. Investigations of general DAEs (cf. [5], [1], [8]) indicate that this is usually the case if the DAE has a higher index. A detailed analysis ([6], [7]) of numerous examples shows that the index may be high and that it depends on different aspects:

- on the formulation of the network equation,
- on the kind of the used network elements,
- on the structure of the circuit,
- on parameter values,
- on operating conditions of the circuit.

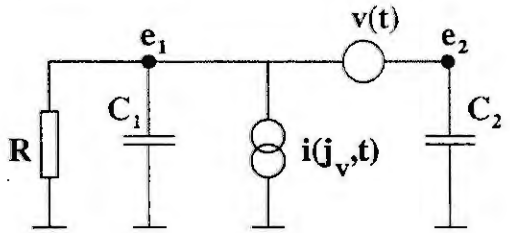
Assuming all (possibly multi-port) capacitances, inductances and resistances to be positive definite as well as certain structural conditions for the controlled sources to be satisfied, the following locally controllable index characterization is possible ([4]):

Theorem 2.1 *The conventional MNA as well as the charge oriented MNA leads to an index-1 DAE if and only if the network contains neither L-I cutsets nor C-V loops. Otherwise, they lead to an index-2 DAE.*

Remark: An *L-I cutset* is a cutset consisting of inductances and/or current sources only. A *C-V loop* is a loop consisting of capacitances and voltage sources only.

3 Consistent initial values

The nonlinear equations in DAE systems represent constraints. Under sufficiently smoothness conditions, DAEs can be considered as differential equations on manifolds. This implies that initial values must belong to a certain manifold, i.e., they have to be consistent. To provide consistent initial values in practice is a nontrivial task but very important for the reliability of the simulation results as the following example (formulated in charge oriented MNA) shows.



$$
\begin{aligned}
q_1' + e_1 + (2\sin(t) + 4)jv - \sin(t) - 2 &= 0 && (5)\\
q_2' - jv &= 0 && (6)\\
e_1 - e_2 &= 2\sin(t) && (7)\\
q_1 &= e_1^2 && (8)\\
q_2 &= e_2 && (9)
\end{aligned}
$$

If we integrate this circuit example with the trapezoidal rule and start from an inconsistent initial value we get a completely wrong numerical solution (see Figure 1). If we start from a consistent value, the numerical solution coincides with the exact solution. In [2], a cheap algorithm for calculating consistent initial values (using operating points) is presented. It exploits the special circuit structure.

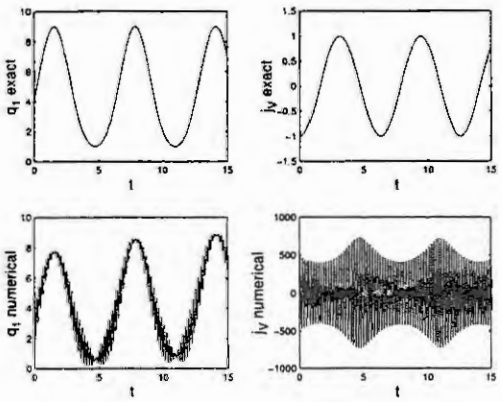
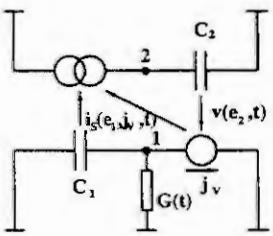


Figure 1: Result for the trapezoidal rule for (5)-(9) starting from the inconsistent value (4,2,2,2,-500)

4 Asymptotic Stability

Numerical methods for DAEs do not always have those stability properties which are well-known for regular ODEs. We want to illustrate this by a simple (theoretically constructed) example.

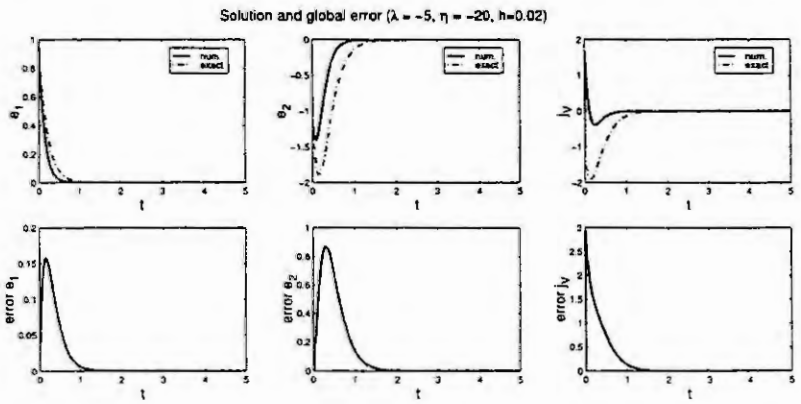


$$\begin{aligned} C_1\dot{e}_1 + G(t)e_1 + j_V &= 0 \\ C_2\dot{e}_2 + i_s(e_1, j_V, t) &= 0 \\ e_1 &= v(e_2, t) \end{aligned}$$

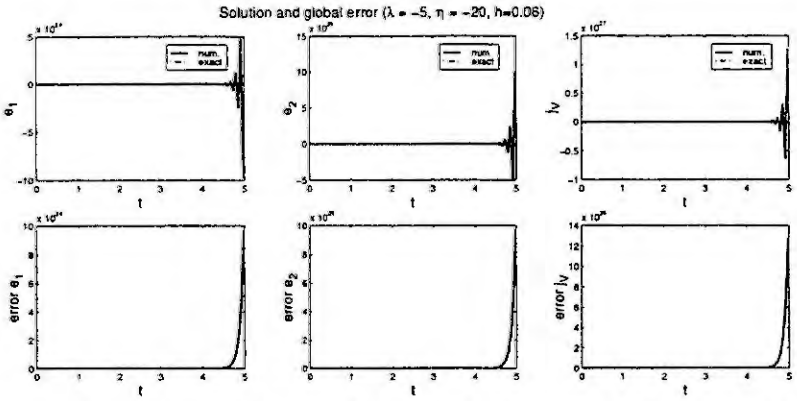
$$C_1 = C_2 = 1,\ G(t) = 1 - \eta t - \lambda,\ v(e_2, t) = -\frac{1}{1-\eta t}e_2$$

$$i_s(e_1, j_V, t) = [\lambda - \eta + (1-\lambda)\eta t - \eta^2 t^2]e_1 + \eta t j_v$$

The implicit Euler method with a stepsize $h = 0.02$ supplies a stable numerical solution as expected.



But if we increase the stepsize slightly ($h = 0.06$) then the numerical solution becomes unstable.



This is an unknown behavior for A-stable methods for regular ODEs and has to be considered when integrating DAEs by standard numerical methods.
Fortunately, this instability effect does not occur if the DAE satisfies certain structural conditions ([9]). In case of integrated circuits, these structural conditions may be characterized in terms of certain modeling criteria ([3]).

References

[1] K. E. Brenan, S. L. Campbell, and L. R. Petzold. *The Numerical Solution of Initial Value Problems in Ordinary Differential-Algebraic Equations.* North Holland Publishing Co., 1989.

[2] D. Estévez Schwarz. Consistent initialization of differential-algebraic equations in circuit simulation. Technical Report 99-5, Fachbereich Mathematik, Humboldt-Univ. zu Berlin, 1999.

[3] D. Estévez Schwarz, A. Rodríguez Santiesteban, and C. Tischendorf. Asymptotic stability in circuit simulation. In preparation.

[4] D. Estévez Schwarz and C. Tischendorf. Structural analysis for electric circuits and consequences for MNA. *Int. J. of Circuit Theory and Applications*, 1999. To appear.

[5] E. Griepentrog and R. März. *Differential-Algebraic Equations and Their Numerical Treatment.* Teubner-Texte zur Mathematik No. 88. BSB B.G. Teubner Verlagsgesellschaft, Leipzig, 1986.

[6] M. Günther and U. Feldmann. CAD based electric modeling in industry. Part I: Mathematical structure and index of network equations. *Surv. Math. Ind.*, 8:97–129, 1999.

[7] M. Günther and U. Feldmann. CAD based electric modeling in industry. Part II: Impact of circuit configurations and parameters. *Surv. Math. Ind.*, 8:131–157, 1999.

[8] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II: Stiff and differential-algebraic problems.* Springer Series in Computational Mathematics 14. Springer-Verlag, Berlin, Heidelberg, 1991.

[9] R. März and A. Rodríguez Santiesteban. Analyzing the stability behaviour of DAE solutions and their approximations. Technical Report 99-2, Fachbereich Mathematik, Humboldt-Univ. zu Berlin, 1999.

NONLINEAR ELECTRICAL NETWORKS AS DYNAMICAL SYSTEMS ON DIFFERENTIABLE MANIFOLDS

Wolfgang Mathis
Otto-von-Guericke-Universität Magdeburg
Institute for Electronics, Signal Processing and Communications
mathis@ipe.et.uni-magdeburg.de/elektronik

Abstract

In this paper we consider the theory of electrical circuits (or networks) from the point of view of differential geometry and present a formulation based on this framework. Furthermore we discuss some applications of this reformulation.

1 Introduction and Historical Remarks

It is known since the early sixties that descriptive equations of electrical circuits – we will denote it as circuit equations in the following – belong to the class of differential equations on differentiable manifolds. This result is related to the celebrated paper of Moser and Brayton [9] in 1964 where their equations for the description of reciprocal and nonlinear circuits are written in coordinates as usual. It lasted another few years until the equations of Moser and Brayton were reformulated by Smale [13] by means the framework of modern differential geometry. Further work was done by Matsomoto, Ishiraku and other to refine this approach for describing electrical networks (see e.g. Mathis [4]). On the other hand Sandberg and Gear tried to solve the socalled "time-constant problem of circuit simulation that was one of the big obstacles to construct an efficient and general purpose circuit simulator. It was emphasized by Gear that crcuit equations should be considered as algebro-differential equations (DAEs) but it lasted another more than ten years if Linda Petzold – a former Ph.D. student of Gear – found out in 1982 that "DAEs are not ODEs (ODEs: ordinary differential equations). For references and further information see e.g. Mathis [5]. At the beginning of the eighties – approximately twenty years after the invention that circuit equations are of a more general type than ordinary differential equations – it became clear that circuit equation should be considered as differential equations on differentiable manifolds or algebro-differential equations. A detailed presentation of the concept of circuit theory from the point of view of modern differential geometry is included in Robert Hermann's monographs on "Interdisciplinary Mathematics" where the following statement is formulated "Electrical circuits offers prototypes and examples of many sorts of abstract mathematical and physical structures; it is extremely useful and important to sort out such generalizations, since it seems that many situations – in biology, chemistry, economics and physics – can be modelled by means of these mathematical structures."

In this paper we will consider some theoretical aspects of circuit equations and discuss its numerical conclusions where we will use ideas from Hermann's presentation.

2 Circuit Theory and Differential Geometry

In order to describe electrical circuits in a sophisticated manner we have to assign the spaces of currents and voltages.

Definition:

Currents: Let be $\mathbb{R}^b$ a real and b-dimensional vector space (column vectors) then a linear subspace $\mathcal{I} \subset \mathbb{R}^b$ is called the space of currents.

Voltages: Let be the ordered pair $(\mathbb{R}^b, \mathcal{I})$ the extended space of currents, $\mathbb{R}^{b^*}$ the dual space of $\mathbb{R}^b$ (row vectors) and let $\mathcal{V} := \mathcal{I}^{orth} \subset \mathbb{R}^{b^*}$ where $\mathcal{I}^{orth}$ is the perpendicular space of $\mathcal{I}$ such that $v(\mathcal{I}) = 0$ (for all $v \in \mathcal{V}$) then $\mathcal{V}$ is called the space of voltages and $(\mathbb{R}^{b^*}, \mathcal{V})$ the extended space of voltages.

Remarks: Of course we have $\dim \mathcal{I} + \dim \mathcal{V} = \dim \mathbb{R}^b = \dim \mathbb{R}^{b^*}$ The aim of the theory of electrical circuits and in essential circuit analysis is to present mathematical structures which are related to physical models of electrical circuits. Using these structures circuit theorists are interested to define certain curves $: t \mapsto (i(t), v(t))$ $(i \in \mathcal{I}, u \in \mathcal{V})$ that may be identified with the physical and time varying currents and voltages of a real electrical circuit.

We will follow ideas from Hermann to present circuit theory in an abstract differential-geometric setting that are presented in the monographs mentioned above. It is known that Kirchhoffs laws form a basic structure of circuit theory. Therefore we will reformulate it in a differential-geometric way. For this purpose we define so-called contact form where we use differential forms in the sense of Cartan.

Let be the ordered pair $(\mathbb{R}^b, \mathcal{I})$ the extended space of currents and $(\mathbb{R}^{b^*}, \mathcal{V})$ the extended space of voltages. Furthermore we suppose that $\{e_i\}_{i=1}^b$ is any basis of $\mathbb{R}^b$ and let $\{\hat{e}_i\}_{i=1}^b$ be the (dual) of $\mathbb{R}^{b^*}$ where $\hat{e}_i(e_i) = \delta_{ij}$. The coordinate functions $x_i : \mathbb{R}^b \to \mathbb{R}$ as well as $y_i : \mathbb{R}^{b^*} \to \mathbb{R}$ can be used to represent any $v \in \mathbb{R}^b$ and $\hat{v} \in \mathbb{R}^{b^*}$, respectively:

$$v = \sum_{i=1}^{b} x_i(v)\, e_i, \qquad \hat{v} = \sum_{i=1}^{b} y_i(\hat{v})\, \hat{e}_i. \tag{1}$$

Based on these definitions we define a contact form Θ on $\mathbb{R}^b \times \mathbb{R}^{b^*}$ as follows $\Theta := \sum_{i=1}^{b} y_i\, dx_i$.

Remarks: It can be shown that this contact form does not on the special choice of the basis in $\mathbb{R}^b$ and $\mathbb{R}^{b^*}$. If we identify $\mathbb{R}^b \times \mathbb{R}^{b^*}$ with the cotangent bundle $\mathcal{T}^*(\mathbb{R}^b)$ on $\mathbb{R}^b$ we have a contact form Θ on $\mathcal{T}^*(\mathbb{R}^b)$. With the following theorem an abstract formulation of Kirchhoff laws for currents and voltages can be given.

Theorem:
Let be $(\mathbb{R}^b, \mathcal{I})$ and $(\mathbb{R}^{b^*}, \mathcal{V})$ the extended spaces of currents and voltages where $\mathcal{V} = \mathcal{I}^{orth}$. Then the contact form Θ is zero on $\mathcal{I} \times \mathcal{V}$.

Definition:
A linear subspace of $\mathbb{R}^b \oplus \mathbb{R}^{b^*}$ whose dimension is equal to the dimension of $\mathbb{R}^b$ – that is b – on which Θ is zero, is called a Kirchhoff subspace of $\mathbb{R}^b \times \mathbb{R}^{b^*}$.

Remarks: In concrete circuits the sets of currents $\mathcal{I}$ and voltages $\mathcal{V}$ are solution sets of homogenous algebraic equations where the coefficients are incidence matrices of a network graph or the transfer coefficients of ideal transformers (see Mathis [4]). It can be shown that these sets are orthogonal if the coefficient matrices form an exact pair of matrices. This property was defined for the case of circuits with a network graph by Ghenzi [1] and generalized by Mathis and Marten [7] to such circuits where the network elements are connected by ideal transformers (connection elements). Note that the existence of the contact structure is related to the theorem of Weyl and Tellegen (see Mathis [4]).

If the coordinate functions x_i of $\mathbb{R}^b$ and y_i of $\mathbb{R}^{b^*}$ of correspond with the currents and voltages of the connection element with n ports then a set of 0-forms and 1-forms can be used to characterize the relationships of currents and voltages at these ports. These forms will be equivalent to the constitutive relations of the classical circuit theory. The forms depend on whether the port is connected with a resistor capacitor, current source or voltage source. We have: Linear resistor: $\Theta_R^0 := v - Ri$; Capacitor: $\Theta_C^1 := C\,dv - i\,dt$; Inductor: $\Theta_L^1 := L\,di - u\,dt$; Current source: $\Theta_I^0 := i - f_I(t)$; Voltage source: $\Theta_U^0 := v - f_U(t)$, where the extended space of currents and voltages as well as the time $\mathbb{R}^b \times \mathbb{R}^{b^*} \times \mathbb{R}_T$ is considered and the coordinate functions are denoted by i or v. If we would like to specify a certain port a subscript with the number of the port is applied. The superscripts "0 or "1 for the forms are used to characterize the order of the forms.

Remarks: In Hermann's monograph the set of these 0-forms and 1-forms is called an ideal (in an algebraic sense). Furthermore you will find a very general mathematical definition of electrical circuits. A generalized circuit systems is a quintuple (M, I, φ, V, K) consisting of a manifold $\mathcal{M}$, an ideal $I_{\mathcal{M}}$ of differential forms on $\mathcal{M}$, called the constitutive ideal, a map $\varphi : \mathcal{M} \to V \times V^*$, and a Kirchhoff subspace $\mathcal{K} \subset V \times V^*$.

Given such an object, set: $\mathcal{M}_K = \varphi^{-1}(K)$ and $I_K := I$ restricted to $\mathcal{M}_K$. A curve $\zeta : t \mapsto \mathcal{M}_K$ is then a trajectory of the system if it is an integral curve of the restricted ideal I_K.

In certain cases another mathematical structure can be helpful for the classification of the set of resistive constitutive relations. This set is related to all above listed 0-forms which have the property that the classical representations with functions do not involve the derivatives of currents and voltages. Therefore we have $F_R(i_1,\ldots,i_b,v_1,\ldots,v_b)=0$ that determine a submanifold of $\mathbb{R}^b \times \mathbb{R}^{b^*}$. Note that we consider only such circuits where the independent current and voltage sources are constant.

It can be shown that the exterior derivative $d\Theta$ is a so-called symplectic structure on $\mathbb{R}^b \times \mathbb{R}^{b^*}$. By means of this 2-form $d\Theta$ the set of zeros of F_R can be classified under certain conditions.

Let the zeros of F_R determine a submanifold $\mathcal{Z}$ of dimension b of $\mathbb{R}^b \times \mathbb{R}^{b^*}$. Furthermore the closed 2-form Θ that forms a symplectic structure on $\mathbb{R}^b \times \mathbb{R}^{b^*}$ is zero on the submanifold $\mathcal{Z}$. If these conditions are satisfied then the resistive constitutive relations are called reciprocal. Note: It is known that $d(d\Theta)$ vanishes identically.

It was observed by Moser [8] that the property of reciprocity is crucial for the formulation of a theory of nonlinear electrical circuits including dissipation of energy. His idea was generalized by Brayton and Moser [9] in 1964 where the differential equations for the class of nonlinear reciprocal circuits were formulated in a systematic manner. From a differential-geometric point of view further mathematical structures have to be defined based on the constitutive relations for the network elements. This was done by Smale [13] in 1972 for the first time and generalized by Matsumoto to a more general class of nonlinear circuits in 1975.

The dynamics of a system and an electrical circuit can be formulated by a set of nonlinear differential equations with respect to currents and voltages where certain algebraic constraints have to be added. From the classical point of view this means that the collection of differential equations for the capacitors and inductors (constitutive relations) have to be combined with Kirchhoff's laws (homogeneous linear algebraic equations) and the resistive constitutive relations (nonlinear equations). In the framework of differential geometry we have to consider (nonlinear) differential equations on the state space that is endowed in generic cases with the structure of a differentiable manifold. Therefore we have to construct the state space $\mathcal{S}$ and the vector field X in order to define the dynamics of a circuit: $\dot{\xi} = X \circ \xi \quad X : \mathcal{S} \to T\mathcal{S}$. The state space $\mathcal{S}$ can be constructed using the Kirchhoff space $\mathcal{K} \subset \mathcal{R}^{l} \times \mathcal{R}^{l^*}$ and the set of zeros of F_R. If the condition of transversality of these subsets is fullfilled their intersection is a differentiable manifold (Smale [13]). For the construction of the vector field X a 2-tensor $g(\cdot,\cdot) : \mathcal{S} \to T*\mathcal{S} \otimes T*\mathcal{S}$ and a 1-form $\omega : \mathcal{S} \to T*\mathcal{S}$ can be obtained from the constitutive relations and Kirchhoff's laws. It was shown by Brayton and Moser [9] that ω can be obtained from a so-called mixed potential P using the exterior derivative, that is $\omega = dP$. We get the 2-tensor g if a 2-tensor G is defined on the linear subspace of inductor currents and capacitor voltages

$$G := \sum_k L(i_L^k) di_L^k \otimes di_L^k - \sum_k C(u_C^k) du_C^k \otimes du_C^k \tag{2}$$

and pullback this 2-tensor on the state space, that is $g = \pi^* G$ where π is a projection from $\mathcal{S}$ to the linear subspace of inductor currents and capacitor voltages. In the same manner the 1-form ω can be obtained. Using these objects an abstract equation for the vector field X can be formulated $g(X,Y) = \omega(Y)$ for all $Y \in T\mathcal{S}$. If this equation has a unique (local) solution the case of a (local) generic circuit dynamics is characterize. The condition for the (local) existence of X is that g is non-degenerated that is if G is non-generated and π^* exists. These conditions can be translated in a more concrete manner and it can be shown that with a suitable "disturbance of the constitutive relations the two conditions are fullfilled - this is called generic.

3 Some Conclusions

Based on the theory of electrical circuits that is discussed in the last section it is find out that solutions of the descriptive equations of this class of physical systems only exists in a unique manner if the 2-tensor g is non-degenerated. The first part of conditions is related to cases where the coefficients (capacitors C and inductors L) are nonzero. If the capacitors and/or inductors are nonlinear these cases appear also for certain times t_0 where $C\left(u_C(t_0)\right) = 0$ and/or $L\left(i_L(t_0)\right) = 0$. Furthermore g can be degenerated if there are difficulties with π^*. A first problem is related to meshes of capacitors and independent voltage sources and/or cutsets of inductors and independent current sources. In these cases the projection π onto the linear subspace of currents and voltages of capacitors and inductors has singularities. But even if no

such cases occur we run into problems if there are nonlinear resistors in the meshes and/or the cutsets that become infinity for certain times t_0. Known applications of this kind of defects are digital circuits and relaxation oscillators where the dynamics take place in an area of the state space S with a fold. If the trajectory of the circuit dynamics reaches a boundary such area a jump occurs and the trajectory proceeds on another piece of S. This subject is discussed in the book of Mishchenko and Rozov [10]) and in the paper of Sastry and Desoer [11] using singular perturbation theory. The noise behavior of circuits and systems that are described by differential equations on differentiable manifolds is discussed by Sastry [12].

From the numerical point of view differential equations on differentiable manifolds is discussed in the framework of algebro-differential equations. A main subject is the so-called index where different concepts were published (see e.g. Mathis [5]).

If we consider the restricted class of reciprocal nonlinear networks it was shown by Moser [8] (see also the paper of Brayton and Moser[9]) that the associated circuit behavior is by a gradient dynamics. Unfortunately it is a gradient with respect to a pseudo-Riemannian metric and therefore there cases with a rather complicate dynamics. First results are contained in the papers of Brayton and Moser [9] and Smale [13] but more recently Larsen [3] discuss some additional aspects of generalized gradient systems of this kind. A relation between descriptive equations of circuits and algebro-differential equations (incl. the index) was discussed by Marten, Chua and Mathis [6] in 1992.

References

[1] Ghenzi, A.G.: Studien über algebraische Grundlagen der Theorie der elektrischen Netzwerke. Dissertation Thesis, Zürich 1953

[2] Hermann, R.: Interdisciplinary Mathematics. Geometric Structure Theory of Systems Control Theory and Physsics. Part A. Brookline, MA: Math. Sci., 1974, Vol. IX.

[3] Larsen, J.C.: On gradient dynamical systems on semi-Riemannian manifolds. JCP 4 (1989) pp. 517-535 (see also: Larsen, J.C.: Electrical Network Theory on Countable Graphs. Trans. Circuits and Systems - 1: Fundm. Theory and Appl. CAS-44 (1997) pp. 1045-1055

[4] Mathis, W.: Theorie nichtlinearer Netzwerke. Springer Verlag, Berlin- New York 1987

[5] Mathis, W.: Recent Developments in Numerical Integration of Differential Equations. Intern. Journ. Num. Mod. 7(1994) pp. 99-125

[6] Marten, W.; L.O. Chua; W. Mathis: On the geometrical meaning of pseudo hybrid and mixed potential. Intern. Journ. Electron. Commun. (AEÜ), 46 (1992) pp. 305-309

[7] Mathis, W.; W. Marten: A Unified Theory of Electrical Networks. Proc. 29th Midwest Symposium of Circuits and Systems, North Holland, Amsterdam 1987

[8] Moser, J.K.: Bistable Systems of Differential Equations with Applications to Tunnel Diode Circuits. IBM Journ. Research Dev. 5(1961) pp. 226-240

[9] Brayton, R.K.; J.K. Moser: A Theory of Nonlinear Networks I+II. Quartly Appl. Math. 22 (1964) pp. 1-33, 81-104

[10] Mishchenko, E.F.; N.K. Rozov: Differential Equations with Small Parameters and Relaxation Oscillations. Plenum Press, New York - London 1980

[11] Sastry, S.S; C. A. Desoer: Jump behavior of circuits and systems. Trans. Circuits and Systems CAS-28 (1981) pp. 1109-1124

[12] Sastry, S.S: The effects of small noise on implicitly defined nonlinear dynamical systems. Trans. Circuits and Systems CAS-30 (1983) pp. 651-663

[13] Smale, S.: On the mathematical foundation of electrical circuit theory. Journ. Differential Geometry 7(1972) pp. 193-210

SUBSTITUTE EQUATIONS FOR INDEX REDUCTION AND DISCONTINUITY HANDLING

G. Fábián[1] **D.A. van Beek**[2] **J.E. Rooda**[3]
Eindhoven University of Technology, Department of Mechanical Engineering
P.O. Box 513, 5600 MB Eindhoven, The Netherlands
E-mail: {d.a.v.beek[2], g.fabian[1], j.e.rooda[3]}@tue.nl

Abstract Several techniques exist for index reduction and consistent initialization of higher index DAEs. Many such techniques change the original set of equations by differentiation, substitution, and/or introduction of new variables. This paper introduces substitute equations as a new language element. By means of a substitute equation, the value of a continuous variable or its time derivative can be specified by an expression. This expression is evaluated each time that the variable or its time derivative, respectively, is referenced in the model. The advantage of substitute equations is that they enable index reduction and consistent initialization of higher index DAEs without changing the original equations; no existing variables are removed and no new variables are introduced. Substitute equations can also be used to enable the use of general purpose numerical solvers for equations where one or more of the unknowns are discontinuous.

Differential algebraic equations

Many current equation based simulation languages use Differential Algebraic Equations (DAEs) to describe the continuous behaviour of the modelled physical system. DAEs are a set of differential equations with additional algebraic constraints in the form

$$\mathbf{f}(\dot{\mathbf{x}}, \mathbf{x}, \mathbf{y}, t) = \mathbf{0}, \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^n$ is the vector of *differential variables*, $\mathbf{y} \in \mathbb{R}^m$ is the vector of *algebraic variables*, $t \in \mathbb{R}$ is the independent variable and $\mathbf{f} \in \mathbb{R}^{2n+m+1} \rightarrow \mathbb{R}^{n+m}$ is the set of DAEs.

In DAEs, not all variables can be freely initialized. The initial values of variables $\mathbf{x}$, $\dot{\mathbf{x}}$ and $\mathbf{y}$, denoted by $\mathbf{x}(0)$, $\dot{\mathbf{x}}(0)$, $\mathbf{y}(0)$, must satisfy equation (1) at time 0:

$$\mathbf{f}(\dot{\mathbf{x}}(\mathbf{0}), \mathbf{x}(0), \mathbf{y}(0), 0) = \mathbf{0}. \tag{2}$$

In DAEs, in many cases, only the differential variables ($\mathbf{x}(0)$) are initialized. The initial values of the algebraic variables ($\mathbf{y}(0)$) and of the time derivatives of the differential variables ($\dot{\mathbf{x}}(\mathbf{0})$) are then calculated from (2).

DAEs are characterized by their (differential) *index* [5]. The index of equation (1) is m, if m is the smallest number such that the system of equations

$$\begin{aligned} \mathbf{f}(\dot{\mathbf{x}}, \mathbf{x}, \mathbf{y}, t) &= \mathbf{0}, \\ \frac{d\mathbf{f}(\dot{\mathbf{x}}, \mathbf{x}, \mathbf{y}, t)}{dt} &= \mathbf{0}, \\ &\vdots \\ \frac{d^{(m)}\mathbf{f}(\dot{\mathbf{x}}, \mathbf{x}, \mathbf{y}, t)}{d^{(m)}t} &= \mathbf{0}, \end{aligned} \tag{3}$$

can be transformed into an explicit ODE ($\dot{\mathbf{x}} = \mathbf{g}(\mathbf{x}, t)$) by algebraic manipulations. In general, the higher the index, the greater the numerical difficulty one is going to encounter when trying to solve the system numerically. For *higher index DAEs* (systems, with index greater than 1) there is no general purpose stable algorithm. A common feature of higher index DAEs is that there are hidden constraints in the DAEs. These are equations that further restrict the initialization of equation (1). Hidden constraints can be obtained after differentiation and algebraic manipulations [8]. The presence of hidden constraints mean that not all differential variables may be chosen freely; there are dependencies among differential variables. This can be seen, if equations in the form

$$\mathbf{g}(\mathbf{x}, \mathbf{u}, t) = \mathbf{0} \tag{4a}$$

$$\mathbf{u} = \mathbf{h}(t) \tag{4b}$$

are present in equation (1), or can be obtained after differentiations. The differential variables in (4a) are *dependent differential variables*. In fact, hidden constraints may be present in index 1 systems of DAEs too.

It is well known, that the mathematical models of several physical systems have a high differential index. The usual technique is that through differentiation and algebraic manipulations, the index is lowered to 0 or 1, and the resulting system is solved with available ODE or DAE solvers. In the literature, several algorithms can be found for index reduction. From these, the algorithm of Gear and Petzold [6], the constraint stabilization technique of Gear [5], and the algorithm of Bachmann et al. [1] all differentiate (parts of) the system of equations and use substitution. The algorithm of Pantelides [8] to reveal hidden constraints in DAEs, can also be used for index reduction. Furthermore, Mattsson describes an index reduction technique in [7], which uses dummy algebraic variables.

There are simulators, where (some of) the above mentioned index reduction techniques are implemented. After the model of a physical system has been specified, the equations are analyzed symbolically and the index is reduced by subsequent steps of differentiations and algebraic manipulations (substitution). In this way, the equations are changed. Different ways of index reduction may thus lead to different sets of variables that can be initialized. This may lead to a modelling problem. Since not all differential variables may be freely chosen in such a system, it must be clear for the modeller which ones may be initialized, and which ones are calculated from the equations. Even more, the modeller may want to choose himself the dependent differential variables that he wants to initialize.

In those simulators, where index reduction is not implemented, only low index systems (index 0 or 1) of equations may be entered. Therefore, the modeller has to perform index reduction, and has to re-formulate the system of equations. In this case, a new equation set is obtained that is usually less expressive than the original one. Also, variables may be eliminated from the equations due to index reduction. Therefore, each time the values of these variables are needed in the model, they must be re-calculated. This also reduces the readability of models.

To overcome the problems of the two approaches, in the χ language [10, 4] substitute equations are used.

Substitute equations

In the χ language, substitution can be specified explicitly by means of *substitute equations* in two forms. The simple form is

$$\begin{aligned} S &::= v \leftarrow E \mid v' \leftarrow E \\ E &::= e \end{aligned}$$

where S and E are nonterminals, v is a continuous variable, v' is the time derivative of a continuous variable and e is a numerical expression. This specification is equivalent to replacing all occurrences of v (v') by e in the model. The guarded form is

$$E ::= [\, b_1 \longrightarrow e_1 \,[]\, \ldots \,[]\, b_n \longrightarrow e_n]$$

where b_i is a boolean guard and e_i is a numerical expression ($i = 1 \ldots n$). In this case, variable v (v') is substituted dynamically, depending on the values of the guards. If b_i is true, variable v (v') is substituted by e_i. If more guards are true at the same time, one alternative is chosen nondeterministically and all occurrences of v (v') are calculated from this alternative.

All variables occurring in the right-hand-side of substitute equations (in expressions e, e_i and b_i, $i = 1 \ldots n$) must be well-defined, either by another substitute equation or by normal, non-substitute equations. Substitute equations are evaluated recursively; if a substituted variable occurs on the right-hand-side of a substitute equation, first, its value is re-calculated by substitution. Therefore, substitute equations can be specified in arbitrary order; the only requirement is that they may not contain circular dependencies. Variables defined in this way may not be assigned.

Substitution facilitates index reduction; it can also be used to reveal hidden constraints in index 1 DAEs and to model discontinuities. This is illustrated below.

Index reduction

As an example for a higher index DAE, take the following PID (proportional integral differential) controller. A horizontal force F is applied to a body of mass m on a flat surface, without friction. The position of the body is

denoted by x. The control objective is to keep the body at a given position x_{set}. The unknowns are x, v, i, e, u. Variable F is an input variable (depending only on time), x_{set}, m, k_P, k_D and k_I are constants.

$$\dot{x} = v \tag{5a}$$
$$\dot{v} = \frac{F-u}{m} \tag{5b}$$
$$\dot{i} = e \tag{5c}$$
$$e = x - x_{set} \tag{5d}$$
$$u = k_P e + k_D \dot{e} + k_I i \tag{5e}$$

This is an index 2 system of DAEs. Differentiation of equation (5d) yields

$$\dot{e} = \dot{x}. \tag{6}$$

After differentiating equations (5a, 5e), and differentiating equation (6) a second time, and then substituting in $\dot{u} = k_P \dot{e} + k_D \ddot{e} + k_I \dot{i}$: v for $\dot{e}$, $\frac{F-u}{m}$ for $\ddot{e}$ ($\ddot{e} = \ddot{x} = \dot{v} = \frac{F-u}{m}$), and $x - x_{set}$ for $\dot{i}$; the ODE form can be obtained. Typically to higher index systems, equations (5) contain a hidden constraint: equation (6) must hold at time 0. This is because e and x are dependent differential variables, as can be seen from (5d). Therefore, only one of them can be initialized freely. Index reduction algorithms differ in the way they choose the variables to substitute. By choosing (5d) for differentiation and $\dot{e}$ for substitution, the system is specified in χ as follows

$$\begin{aligned}
& x' = v \\
, \; & v' = (F-u)/m \\
, \; & e = x - x_{set} \\
, \; & i' = e \\
, \; & u = k_P e + k_D e' + k_I i. \\
, \; & e' \leftarrow x'
\end{aligned}$$

Variable e is a so-called *prime substituted differential variable*. Variables of these category cannot be freely initialized. In this model, x can be freely initialized, but the value of e depends on x. The actual set of equations solved by numerical solvers obtained after substitution is

$$\dot{x} = v \tag{7a}$$
$$\dot{v} = \frac{F-u}{m} \tag{7b}$$
$$e = x - x_{set} \tag{7c}$$
$$\dot{i} = e \tag{7d}$$
$$u = k_P e + k_D \dot{x} + k_I i. \tag{7e}$$

Note that for the solvers, $\dot{e}$ is not present in the equations, e is thus an algebraic variable.

The advantage of using substitute equations is that the process of substitution is transparent; the original form of the equations is preserved and the additional information used ($\dot{e} = \dot{x}$) is made explicit. Also, references to the substituted variable in the discrete-event part of the model need not be altered; variable e is by definition a differential variable, and its time derivative can be referenced in any discrete statement.

The index of the example system of equations (5) can also be reduced by removing variable e from the equations. In this case, both e and $\dot{e}$ are calculated by substitution. The χ specification of the equations is as follows

$$\begin{aligned}
& x' = v \\
, \; & v' = (F-u)/m \\
, \; & i' = e \\
, \; & u = k_P e + k_D e' + k_I i. \\
, \; & e \leftarrow x - x_{set} \\
, \; & e' \leftarrow x'
\end{aligned}$$

In this case, variable e is a so-called *differential base-prime substituted* variable. Wherever e and $\dot{e}$ occur in the model they are substituted by the right-hand-side expressions of the respective substitute equations. The equation

set actually solved by numerical solvers obtained by substitution is

$$\begin{aligned} \dot{x} &= v \\ \dot{v} &= \frac{F-u}{m} \\ \dot{i} &= x - x_{set} \\ u &= k_P(x - x_{set}) + k_D\dot{x} + k_I i. \end{aligned}$$

Variable e has disappeared from the actual equations. What is left are four equations in four unknowns. This is an index 1 problem. Again, in this χ specification only the value of x can freely be chosen, after which e is automatically initialized to $x - x_{set}$ via substitution. Also, the value of $\dot{e}$ is set automatically to $\dot{x}$.

Consistent initialization of index 1 systems

In the previous two approaches, the problems of higher index DAEs and hidden constraints in the equations have been solved simultaneously. In this section, the situation is addressed where only the problem of a hidden constraint is present. This is the case when the system of DAEs is of index 1 and there are dependent differential variables. Again, the solution is substitution.

The index of the PID controller example can also be reduced if variable u is replaced by a derivative of a dummy variable z. The set of equations now is

$$\dot{x} = v \tag{8a}$$
$$\dot{v} = \frac{F-\dot{z}}{m} \tag{8b}$$
$$e = x - x_{set} \tag{8c}$$
$$\dot{i} = e \tag{8d}$$
$$\dot{z} = k_P e + k_D\dot{e} + k_I i. \tag{8e}$$

This is an index 1 problem, because after differentiating equation (8c), the equations can be re-arranged into an ODE. Yet, e and x remain dependent differential variables so that they cannot be freely initialized. As a consequence, there is a hidden constraint in equation (8c), which appears after differentiation of the equation. The initialization problem can easily be solved in χ as before, by adding a substitution equation for $\dot{e}$ (or for both e and $\dot{e}$).

Modelling discontinuities

Another application area for substitute equations is the modelling of discontinuous functions. General purpose DAE and ODE solvers cannot usually integrate discontinuous functions [2]. The usual approach is that discontinuities are specified by so called *switching functions*. When the sign of the switching function changes, a discontinuity occurs. Integration stops, and is re-started again in after the discontinuity. For more on numerical methods with respect to discontinuities we refer to [3].

A discontinuity in a variable that is used by the solver can be avoided in cases where the discontinuous variable can be expressed in a closed form. This variable can then be calculated by substitution, and thereby, it is removed from the equation set that is actually solved by numerical solvers. As an example, consider a tank described in [9], where overflow occurs if the level h of its contents reaches a maximum height h_{max}. The incoming and outgoing flows are denoted by Q_i and Q_o, respectively; the area of the tank by A, and the overflow by Q_x. The system described by a conditional equation is

$$\begin{aligned} &[\ h < h_{max} \vee Q_i < Q_o \longrightarrow Ah' = Q_i - Q_o,\ Q_x = 0 \\ &[]\ h \geq h_{max} \wedge Q_i \geq Q_o \longrightarrow Ah' = 0,\ Q_x = Q_i - Q_o \\ &] \end{aligned}$$

The general form of a conditional equation is: $[\, b_1 \longrightarrow DAEs_1\ [] \ldots [] \ b_n \longrightarrow DAEs_n\,]$, where $DAEs_i$ $(1 \leq i \leq n)$ represents one or more DAEs separated by commas. Boolean expression b_i denotes a guard. At any time, (at least) one of these guards must be open (true), so that the DAE(s) associated with the open guard (after the arrow of the open guard) is (are) activated. The discontinuous variable Q_x can be removed from the equations by substitution

$$
\begin{aligned}
& Ah' = Q_i - Q_o - Q_x \\
, \; & Q_x \leftarrow [\; h < h_{max} \vee Q_i < Q_o \longrightarrow 0 \\
& \quad [] \; h \geq h_{max} \wedge Q_i \geq Q_o \longrightarrow Q_i - Q_o \\
& \quad]
\end{aligned}
$$

In this case, only the first equation, $Ah' = Q_i - Q_o - Q_x$ is solved by integration. The fact that variable h has a discontinuous first derivative is usually not a problem for numerical integrators.

Conclusions

Substitute equations make the mechanism of index reduction transparent to users. The original equation set is unchanged, so that substituted variables do not disappear from the model; they can still be used in discrete-event statements. In this way, expressiveness of the models is preserved. Furthermore, the use of substitute equations makes it clear which variables can be chosen freely, and which ones are calculated. Substitute equations can also be used to reveal hidden constraints in index 1 DAEs. Finally, substitute equations enable the use of general purpose numerical solvers for equations where an unknown variable is discontinuous and can be expressed in a closed form.

References

1. Bachmann, R., Brüll, L., Mrziglod, T., and Pallaske, U., On methods for reducing the index of differential algebraic equations. Computers and Chemical Engineering, 14 (1990), 1271 – 1273.
2. Cellier, F. E., Elmqvist, H., Otter, M., and Taylor, J. H., Guidelines for modeling and simulation of hybrid systems. In: IFAC 12th Triennial World Congress, Sydney, 1993, 1219 – 1225.
3. Eich-Soellner, E. and Fuehrer, C., Numerical Methods in Multibody Dynamics. Teubner, Stuttgart, 1998.
4. Fábián, G., A Language and Simulator for Hybrid Systems. Ph.D. thesis, Eindhoven University of Technology, 1999.
5. Gear, C. W., Differential-algebraic equation index transformations. SIAM. J. Sci. Stat. Comp., 9 (1988), 39 – 47.
6. Gear, C. W. and Petzold, L. R., ODE methods for the solution of differential/algebraic systems. SIAM Journal on Numerical Analysis, 21 (1984), 716 – 728.
7. Mattsson, S. E. and Söderlind, G., Index reduction in differential-algebraic equations using dummy derivatives. SIAM J. Sci. Comput., 14 (1993), 677 – 692.
8. Pantelides, C. C., The consistent initialization of differential-algebraic systems. SIAM. J. Sci. Stat. Comput., 9 (1988), 213 – 231.
9. Van Beek, D. A. and Rooda, J. E., Specification of discontinuities in hybrid models. In: Hybrid Dynamical Systems—Proc. of 3rd International Conference on Automation of Mixed Processes, Reims, 1998, 415 – 420.
10. Van Beek, D. A. and Rooda, J. E., Languages and applications in hybrid modelling and simulation: Positioning of Chi. Control Engineering Practice, 8 (2000), 81 – 91.

SYMBOLICALLY CALCULATED HIGHER INDEX CONDITIONS FOR LINEAR CIRCUITS

C. Clauß, P. Schwarz, B. Straube, W. Vermeiren
Fraunhofer-Institute Integrierte Schaltungen, Design Automation Department (EAS) Dresden
Zeunerstraße 38, D-01069 Dresden, Germany
Tel.: +49-351-4640 737 e-mail: (clauss, schwarz, straube, vermeire)@eas.iis.fhg.de

Abstract. For linear dynamic circuits the condition of the index to be higher than one is calculated symbolically. Using Analog Insydes a function for the computer algebra system Mathematica is written. It calculates the index condition if the sparse tableau analysis method is applied to the circuit. Examples illustrate the performance of the method.

Introduction

Higher index differential-algebraic equations (DAE's) are severe challenges for simulation tools. Often simulations fail [1], or they run to completely false results in some cases (e.g. [2]). There are two possibilities to overcome these problems: Either a tool is chosen which is capable of dealing with higher index DAE's, or the DAE is modified to get a lower index. Both directions are fields of research [3][4].

One of the key questions is the determination of the index. There are several index concepts which coincide in the case of linear DAE's. Linearity assumed, the index calculation is theoretically well understood [5]. For linear circuits different methods are under investigation, which calculate the DAE index utilizing structural information under restricted circuit element conditions [6].

In this paper a simple method is described which derives a condition about the higher index of the DAE which is constructed for any linear circuit via sparse tableau analysis (STA). Using a symbolic circuit analysis tool (Analog Insydes [7]) the condition of a higher index (index > 1) of the tractability index concept [8] is calculated symbolically. The condition is applicable to

- the decision whether the DAE of a given circuit is a higher index one,
- finding out how circuit element parameters have to be chosen to get/avoid a higher index.

The higher index condition

We consider a linear DAE with constant coefficient matrices (e.g. a descriptor form, A can be singular)

(1) $$Ax'(t) + Bx(t) + f(t) = 0$$

($x(t)$, $f(t)$: $T \to R^m$, $A, B \in R^{m \times m}$, $T \subseteq R^1$ time interval, $m>0$). According to the tractability index definition [8] the following matrix chains have to be calculated:

(2.1) $$A_0 = A$$

(2.2) $$B_0 = B$$

(2.3) $$A_{i+1} = A_i + B_i Q_i$$

(2.4) $$B_{i+1} = B_i P_i$$

with $P_i = I - Q_i$, Q_i projector on the kernel $ker\, A_i$, i=0, 1, ... The tractability index is the number τ with A_τ regular and A_j singular for all $j<\tau$.
Consequently, the index is higher than one if and only if A_1 is singular. Therefore, the condition

(3) $$det(A + BQ_0) = 0$$

has to be evaluated. To get this condition symbolic circuit analysis is used.

Symbolic circuit analysis

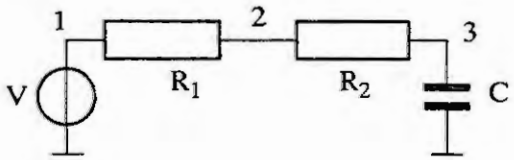
Analog Insydes [7] is a program for the symbolical analysing and sizing analog electronic circuits. It is based on the computer algebra system Mathematica [9] utilizing its symbolic calculation capabilities. For handling circuit analysis a set of Mathematica functions is defined. We use Analog Insydes to derive (1) from a circuit description. The further calculation steps use Mathematica functions only. The higher index condition (3) is derived by the following steps:

- Describe the circuit in an element oriented way
  There is an input format which uses the Mathematica list capabilities. A netlist converter for reading SPICE netlists is available.
- Set up the STA equations
  It is possible using the 'CircuitEquations' analysis function with the 'SparseTableau' option.
- Extract the A and B matrices
  This is possible using Mathematica functions.
- Construct the projector Q_0 on $ker\, A$
  At the moment this is restricted to the usage of STA. Concerning capacitors and inductors the projector can be constructed by determining the zero rows of the matrix A.
- Calculate the determinant, and factorize it
  The matrix $A_1=A+BQ_0$ is calculated. Symbolic determinant calculation and factorization are functions of the Mathematica system.

The symbolic analysis steps are combined to a user defined Mathematica function.

Examples

The following example gives a short impression of the usage of the calculation of the higher index condition. It is



a very simple example only to demonstrate the calculation steps. The Analog Insydes netlist of this RC circuit is:

```
Net = Circuit[
  Netlist[
    {V,  {1,0},  V},
    {R1, {1, 2}, R1},
    {R2, {2, 3}, R2},
    {C,  {3, 0}, C}
]]
```

The generated matrices A and B are:

$$(4)\qquad A = \begin{bmatrix} 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&C&0&0&0&0 \end{bmatrix} \qquad B = \begin{bmatrix} -1&1&1&1&0&0&0&0\\ 0&0&0&0&1&1&0&0\\ 0&0&0&0&0&-1&1&0\\ 0&0&0&0&0&0&-1&1\\ 1&0&0&0&0&0&0&0\\ 0&-1&0&0&0&R_1&0&0\\ 0&0&-1&0&0&0&R_2&0\\ 0&0&0&0&0&0&0&-1 \end{bmatrix}$$

The projector on the kernel of A is:

$$(5)\qquad Q_0 = diag(1,1,1,0,1,1,1,1)$$

The calculated higher index condition obtained by Mathematica is

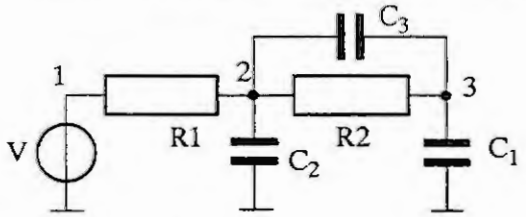
$$(6)\qquad C(R_1 + R_2) = 0$$

If this condition is fulfilled the DAE has an index higher than 1. There are two factors in (6), therefore:

- $C=0$: This is not possible, because the construction of the projector presumed C not to be zero.
- $R_1=-R_2$: It leads to a higher index DAE indeed.

In this example a formula was calculated symbolically which allows to determine the circuit element parameters in such a way that the STA equation system is of higher index. Sometimes the determinant of the A_1-matrix is zero independently from special parameter values. Then the index is higher due to the DAE structure [10]. E.g. this can

be observed if a mesh of capacitor branches occurs like in the following example:



The Analog Insydes netlist of this example circuit is:

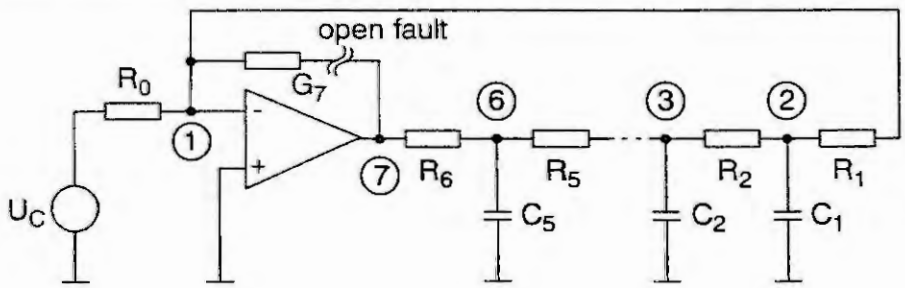
```
Net = Circuit[
  Netlist[
    {V,  {1,0},  V},
    {R1, {1, 2}, R1},
    {R2, {2, 3}, R2},
    {C1, {3, 0}, C1},
    {C2, {2, 0}, C2},
    {C3, {2, 3}, C3}
]]
```

Without mentioning the intermediate steps the symbolically calculated higher index condition is zero. As expected the index is higher than 1 in any case of parameters R_1, R_2, C_1, C_2, C_3. Therefore, the symbolic higher index condition gives insight into the structural higher index property.

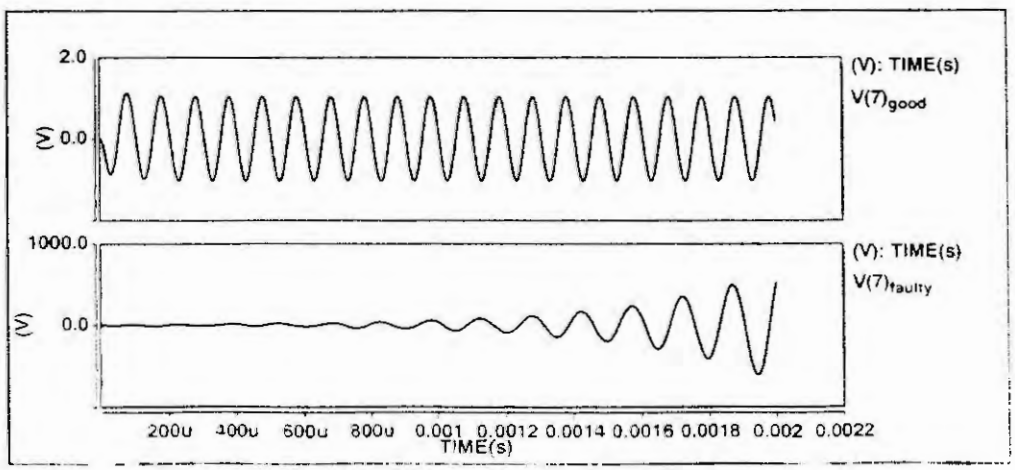
Example for detecting index problems in test signal evaluation

Analogue fault simulation [11] is applied to evaluate test signals for an analogue network for fault detecting. A fault simulator creates a faulty network by *fault injection*. The electric behaviour of a fault is usually modelled by network elements and their interconnections. Commonly, two-poles for *shorts* and *opens* or n-poles, e.g. for defective transistors, are injected into the network. Caused by such a fault injection the electric behaviour of the now faulty network changes.

As an example the RC-network [2] with an operational amplifier depicted in the following picture is considered. If the operational amplifier is modeled ideally by a nullator-norator-pair, the fault free network can be simulated without any problems. In the case of the *open* fault no simulation was possible.



values:
$R_i = 1000$
$C_1, C_2, C_4, C_5 = 1.e\text{-}9$
$C_3 = 1.e\text{-}6$
$U_c = 0.1\sin(20000\pi t)$
$G_7 = .001$



If otherwise the operational amplifier is modeled using a voltage controlled voltage source (amplification V=15000) the fault-free network can be simulated too. Its behaviour (node 7) is shown in the result plot (upper window). If the *open* fault is injected at G_7 the simulator has calculated the behaviour shown in the lower window of the result plot. Note the different scales of the voltage ranges. An analogue fault simulator would report this fault as to be detected. However, there is an index problem. The faulty network has the *index = 6*. Thus, the high index number is the reason for the above mentioned behaviour.

The application of the higher index condition formula method allows

- to detect a higher index already after fault injection but without the fault simulation run.
- to give a warning if in the case of a higher index a fault is reported as to be detected.

On the other hand the conditions for an index increase can be calculated for different description levels of the operational amplifier. In this example assumed an ideal operational amplifier the condition for a higher index is

$$(7) \qquad C_1 \cdot C_2 \cdot C_3 \cdot C_4 \cdot C_5 \cdot G_7 \cdot R_0 \cdot R_1 \cdot R_2 \cdot R_3 \cdot R_4 \cdot R_5 \cdot R_6 = 0$$

That means the index increases if $G_7=0$ which is the open fault, or if $R_i=0$ $(i=0,...6)$ which leads to meshes of capacitors. Assumed the operational amplifier is represented by a voltage controlled voltage source the higher index condition changes into

$$(8) \qquad C_1 \cdot \ldots \cdot C_5 \cdot R_2 \cdot \ldots \cdot R_6(R_0 + R_1 + G_7 \cdot R_0 \cdot R_1 - G_7 \cdot R_0 \cdot R_1 \cdot V) = 0$$

The index increases if:

$$(9) \qquad G_7 = (R_0 + R_1)/(R_1 \cdot R_1 \cdot (V - 1))$$

In the case of the open fault the network can be simulated because of the index which is not high. But the result is doubtful because G_7 is in the critical region according to (9) if G_7 comes closer to zero. The symbolic higher index condition gives a useful insight.

Conclusion

A simple method for the symbolic calculation of the higher index condition for linear circuits is presented. It is quite useful both for the recognition and for the search of higher index DAE's (e.g. for simulations). Due to the symbolic calculation the method seems to be more robust than pure numerical calculations [12]. The restrictive presumptions (STA only, constant matrices) should be overcome by further investigations. Exploiting other matrix formulation methods than STA is very important because the index is not related to the physical system itself but to the method used for the formulation of the systems equations. The performance of the method is demonstrated.

References

1. Petzold, L., Differential/algebraic equations are not ODEs. SIAM J. Sci. Stat. Comput., 3 (1982), 367-384.
2. Straube, B., Reinschke, K. et al., On the fault-injection-caused increase of the DAE-index in analogue fault simulation. IEEE European Test Workshop ETW'99, Constance, Germany, May 25.-28, 1999, 118-122.
3. Brenan, K. E., Campbell, S. L. and Petzold, L. R., Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations. North-Holland, New York, 1989.
4. Hairer, E. and Wanner, G., Solving Ordinary Differential Equations II: Stiff and Differential-algebraic Problems. Springer-Verlag 1991.
5. Gantmacher, F. R., Matrizentheorie II. Deutscher Verlag der Wissenschaften, Berlin, 1959.
6. Röbenack, K. and Reinschke, K., Graph-theoretically determined Jordan block size structure of regular matrix pencils. Linear Algebra Appl. 263 (1997), 333-348.
7. Hennig, E. and Halfmann, T., Analog Insydes Tutorial. ITWM, Kaiserslautern, 1998.
8. März, R., Numerical methods for differential-algebraic equations. Acta Numerica, 1991, 141-198.
9. Wolfram, S., Mathematica - Ein System für Mathematik auf dem Computer. Addison-Wesley Publishing Company, 1992.
10. Tischendorf, C., Topological index calculation of DAEs in circuit simulation. Surv. Math. Ind., 8(3-4)1999, 187-199.
11. Straube, B., Müller, B., Vermeiren, W., Hoffmann, C. and Sattler, S., Analogue Fault Simulation by aFSIM. Paper Accepted for Presentation at: DATE'00, User Forum, Paris, March 27-30, 2000.
12. Matz, K. and Clauß, C., Simulation support by index computation. 15th IMACS World Congress, Berlin, Aug. 1997, Vol. 1, 203-208.

A further index concept for linear PDAEs of hyperbolic type

Yvonne Wagner
Technische Universität Darmstadt, FB Mathematik
Schloßgartenstr. 7, D-64289 Darmstadt, Germany

Abstract: For many technical systems the use of a refined network approach yields mathematical models given by initial-boundary value problems of partial differential algebraic equations (PDAEs). The boundary conditions of these systems are governed by time-dependent differential-algebraic equations (DAEs) that couple the PDAE system with the network elements that are modelled by DAEs in time only. As the numerical difficulties for a DAE can be classified by the index concept, it seems to be natural to generalize these ideas to the PDAE case. There already exist some approaches for parabolic and hyperbolic equations [1, 6, 4]. Here we will focus on a new kind of index, the characteristics index for hyperbolic equations, that does not depend on the elimination of one of the independent variables. It relies on the fact that hyperbolic PDEs can be regarded as ODEs along the characteristics.

1 Introduction

In technical simulation of time dependent processes most of today's industrial software is based on a network approach [5]. Only topology, and no spatial dimension is considered. However, if coupling and second order effects become more important or distributed elements have to be considered, the network approach has to be combined with corresponding models to cope with the spatial extension. In several applications [3, 7, 8] one has to deal with mixed systems of differential-algebraic systems (DAEs) in time only and hyperbolic systems of partial differential equations (PDEs) both in time and space. These systems are coupled by appropriate physical boundary conditions, connecting the network variables at the boundaries with the inner variables. For pure DAE systems, the index concept has turned out to give a deep insight into the solution properties, as well as in the numerical problems to be expected when solving these systems. Generalization of the index concept to linear PDAE systems has recently been proposed in [1, 6] with a main focus on parabolic systems. Some of these concepts were transferred and extended to hyperbolic systems in [4]. A time and space index were defined and compared with the indices of the semidiscretized system. In this work we focus on the characteristics index that is independent of special elimination methods of the time or space variable. Furthermore we will compare with some examples the value of the characteristics index with a perturbation index, that measures the influence of small perturbations in the initial and boundary data on the numerical solution.
Linear hyperbolic-type equations are defined in [4] by

$$Au_t + Bu_x + Cu = f(x,t), \quad u = u(x,t), \quad x \in [0,1], \quad t \in [0,T] \tag{1}$$

where $u, f \in C^1$. In order to guarantee the hyperbolicity we assume A regular and $A^{-1}B$ real diagonalizable. The initial conditions are given by $u(x,0) = g(x)$ and the boundary conditions by the linear DAE in time

$$R_1 \begin{pmatrix} u_t(0,t) \\ u_t(1,t) \\ \dot{z}(t) \end{pmatrix} + R_2 \begin{pmatrix} u(0,t) \\ u(1,t) \\ z(t) \end{pmatrix} - s(t) = 0\,. \tag{2}$$

Hereby z denotes the additional network variables. As we restrict ourselves to the analysis of linear systems of the type (1) and (2), we obtain smooth solutions, if $f(x,t)$, $g(x)$ and $s(t)$ are smooth enough.

2 The characteristics index

Considering the definitions of time and space index given in [4], we find that the t resp. x-variable has to be eliminated first with some special method. In order to avoid this, the hyperbolic PDAE is transformed to an ODE along the characteristics. The transformation of (1) to characteristic form

$$v_t + \Lambda v_x + \tilde{C} v = \tilde{f} \quad \text{with} \quad \Lambda = diag(\lambda_1, \ldots, \lambda_{n_p}), \tag{3}$$

with $\lambda_i > 0$ for $i = 1, \ldots, l$, with $\lambda_i < 0$ for $i = l+1, \ldots, r$ and with $\lambda_i = 0$ for $i = r+1, \ldots, n_p$ gives n_p charakteristic curves $x = k_i^a(t)$ for $i = 1, \ldots, n_p$ with the parameter $a \in \mathbb{R}$. The functions $k_i^a : \mathbb{R} \longrightarrow \mathbb{R}$ are linear. Thus, we get for the system of ODEs along the characteristics

$$\begin{aligned}
\frac{dv_1(k_1^a(t),t)}{dt} &= \sum_{i=1}^{n_p} \tilde{c}_{1i} v_i(k_1^a(t),t) + \tilde{f}_1(k_1^a(t),t)\,, \\
&\vdots \\
\frac{dv_{n_p}(k_{n_p}^a(t),t)}{dt} &= \sum_{i=1}^{n_p} \tilde{c}_{n_p i} v_i(k_{n_p}^a(t),t) + \tilde{f}_{n_p}(k_{n_p}^a(t),t).
\end{aligned}$$

$\tilde{c}_{ji}$ are the entries in the coupling matrix $\tilde{C}$. The initial conditions for these ODEs are given by the initial resp. boundary conditions of the original PDAE. When integrating the system, they are inserted depending on the characteristics. The equations for $1 \le i \le l$ are

$$\begin{aligned}
0 \le k_i^a(0) \quad &: \quad \int_0^t \frac{dv_i(k_i^a(\tau),\tau)}{d\tau} d\tau = v_i(k_i^a(t),t) - g_i(k_i^a(0))\,, \\
0 > k_i^a(0) \quad &: \quad \int_{\tilde{t}}^t \frac{dv_i(k_i^a(\tau),\tau)}{d\tau} d\tau = v_i(k_i^a(t),t) - v_i(0,\tilde{t}) \quad \text{with} \quad k_i^a(\tilde{t}) = 0,
\end{aligned}$$

and those for $l+1 \le i \le r$

$$\begin{aligned}
1 \ge k_i^a(0) \quad &: \quad \int_0^t \frac{dv_i(k_i^a(\tau),\tau)}{d\tau} d\tau = v_i(k_i^a(t),t) - g_i(k_i^a(0))\,, \\
1 < k_i^a(0) \quad &: \quad \int_{\tilde{t}}^t \frac{dv_i(k_i^a(\tau),\tau)}{d\tau} d\tau = v_i(k_i^a(t),t) - v_i(1,\tilde{t}) \quad \text{with} \quad k_i^a(\tilde{t}) = 1.
\end{aligned}$$

The characteristic curves for $r+1 \le i \le n_p$ are constants, i. e. only the initial conditions influence the solution:

$$\int_0^t \frac{dv_i(k_i^a,\tau)}{d\tau} d\tau = v_i(k_i^a,t) - g_i(k_i^a)\,.$$

The values $v_i(0,\tilde{t})$ resp. $v_i(1,\tilde{t})$ are determined by the boundary conditions. If the solution components are coupled at the boundaries, these components have to be computed gradually along the other characteristics until the time layer $t = 0$ is reached. Using this procedure we can define an index that is determined by the boundary values and the influence of the initial conditions. For the PDAE system itself only yields ODEs along the characteristics. In order to obtain a full system for the solutions at the boundaries, we have to complete the system by the integrated ODEs.

Definition: Assume that all equations for the formal determination of the solutions at the boundaries are given, the boundary and initial conditions and the additional ODEs along the characteristics. The network variable z is eliminated. Then the *characteristics index* ν_C denotes the differential index of this system.

Remark: The characteristics index does not depend on a special method to eliminate the space or time variable. However, the DAE system in time is set up with integrals along different characteristic curves. It is shown that the numerical problems are determined mainly by the boundary conditions but that also initial conditions play a role in solving PDAEs of hyperbolic type.

3 Examples

Example 1: Considering purely time-dependent algebraic boundary conditions without coupling or network components, the characteristics index is one, $\nu_C = 1$. This value coincides with the value of the perturbation index as defined in [4]. The characteristics and perturbation index are also the same, if additional network components appear — but then it is possible to obtain a higher index.

Example 2: For $l = 1$, $r = n_p = 2$ and $f \equiv 0$ the boundary values are coupled:

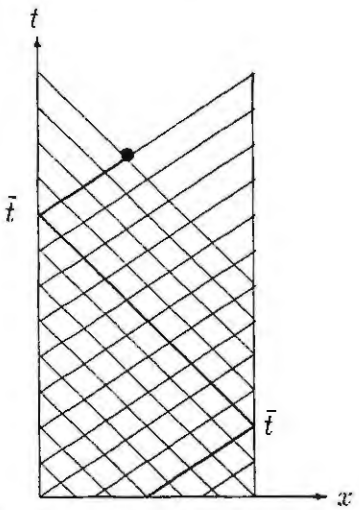
$$v_1(0,t) + v_2(0,t) = s_1(t), \quad v_1(1,t) - v_2(1,t) = s_2(t)\,.$$

If we want to compute the solution v_1 at the marked point (x,t) in the following sketch, we get with $k_1^a(\tilde{t}) = 0$

$$\int_{\tilde{t}}^{t} \frac{dv_1(k_1^a(\tau),\tau)}{d\tau} d\tau = v_1(k_1^a(t),t) - v_1(0,\tilde{t}) = \int_{\tilde{t}}^{t} \tilde{c}_{11} v_1(k_1^a(\tau),\tau) + \tilde{c}_{12} v_2(k_1^a(\tau),\tau) d\tau\,.$$

Evaluation at the left boundary $v_1(0,\tilde{t}) = s_1(\tilde{t}) - v_2(0,\tilde{t})$ yields first the computation of $v_2(0,\tilde{t})$ with $k_2^{\bar{a}}(\tilde{t}) = 0$ and $k_2^{\bar{a}}(\bar{t}) = 1$. This is done by integration

$$\int_{\tilde{t}}^{\bar{t}} \frac{dv_2(k_2^{\bar{a}}(\tau),\tau)}{d\tau} d\tau = v_2(0,\tilde{t}) - v_2(1,\bar{t}) = \int_{\tilde{t}}^{\bar{t}} \tilde{c}_{21} v_1(k_2^{\bar{a}}(\tau),\tau) + \tilde{c}_{22} v_2(k_1^{\bar{a}}(\tau),\tau) d\tau\,.$$



Thus, we reach the opposite boundary along the characteristic line where

$$v_2(1,\bar{t}) = v_1(1,\bar{t}) - s_2(\bar{t})$$

holds. Again $v_1(1,\bar{t})$ is not given, but with another integration along the second characteristic line, we can insert directly the initial conditions, i. e. we get with $k_1^{\bar{a}}(\bar{t}) = 1$

$$v_1(1,\bar{t}) = g_1(k_1^{\bar{a}}(0)) + \int_0^{\bar{t}} \ldots d\tau.$$

Therefore the solution can be expressed on an arbitrarily chosen point with integrals over the source terms along the characteristics.

Combining the above equations to a system for all unknowns, we obtain $\nu_C = 1$ for the characteristics index. This is due to the fact, that the value at the left resp. at the right only influences one boundary condition at a time.

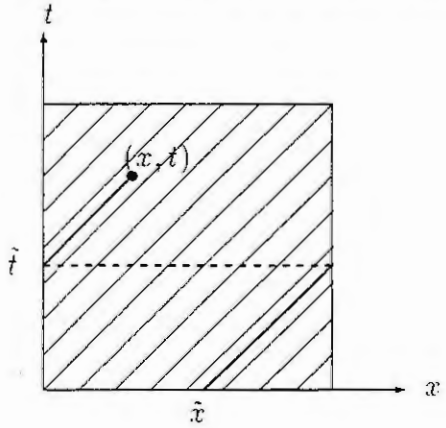
Example 3: We consider two uncoupled advection equations $v_t + v_x = 0$ and $w_t + w_x = 0$ with initial data $v(x,0) = g_1(x)$, $w(x,0) = g_2(x)$ and boundary conditions

$$\begin{aligned} v(0,t) + w(0,t) + v(1,t) &= s_1(t)\,, \\ w(0,t) + w(1,t) &= s_2(t), \end{aligned}$$

where the solution at the right and at the left side appears in the same equation. This system yields for $x < t$ and $t < 1$ the analytical solution

$$\begin{aligned} v(x,t) &= s_1(t-x) - g_1(1-t+x) - s_2(t-x) + g_2(1-t+x)\,, \\ w(x,t) &= s_2(t-x) - g_2(1-t+x)\,. \end{aligned}$$

Thus, the derivatives of both boundary and initial data influences the solution. Measuring the sensitivity of the whole system with the perturbation index ν_P, we get $\nu_P = 2$.



For the characteristics index we have to set up the system for the physically given boundary values and the solution at the other boundary due to the coupling in the boundary conditions. The latter is computed by integration of the ODEs along the characteristics. As illustrated in the sketch at the left, the whole system — four equations in $\tilde{t}$ for four unknowns — thus reads for $x < t < 1$:

$$\begin{aligned} v(0,\tilde{t}) + w(0,\tilde{t}) + v(1,\tilde{t}) &= s_1(\tilde{t}), \\ w(0,\tilde{t}) + w(1,\tilde{t}) &= s_2(\tilde{t}), \\ v(1,\tilde{t}) - g_1(\tilde{x}) &= 0, \\ w(1,\tilde{t}) - g_2(\tilde{x}) &= 0. \end{aligned}$$

This system has differential index 2, $\nu_C = 2$, as we would expect from the exact solution and the perturbation index.

4 Summary

In this paper we have presented a further index concept for linear PDAEs of hyperbolic type. Using the fact that the PDAE can be written as an ODE system along the characteristic curves, we can compute the differential index of this system together with the boundary functions. This approach yields for different examples the same index value as the perturbation index und thus reflects well the sensitivity of the PDAE w.r.t. initial and boundary conditions.

References

[1] S. L. Campbell, W. Marszalek, The Index of an Infinite Dimensional Implicit System. To appear in Mathematical and Computational Modelling of Dynamical Systems.

[2] D. Furihata, private communication. RIMS Kyoto, 1999.

[3] M. Günther, A joint DAE/PDE model for interconnected electrical network. To appear in Mathematical and Computer Modelling of Dynamical Systems.

[4] M. Günther, Y. Wagner, Index concepts for linear mixed systems of differential-algebraic and hyperbolic-type equations. Submitted.

[5] M. Hoschek, P. Rentrop, Y. Wagner, Network Approach and Differential–Algebraic Systems in Technical Applications. To appear in *Surv. Math. Ind.*.

[6] W. Lucht, K. Strehmel, C. Eichler-Liebenow, Indexes and Special Discretization Methods for Linear Partial Differential Algebraic Equations. *BIT*, 39 (1999), No. 3, 484-512, 1999.

[7] B. Simeon, Modelling a Flexible Slider Crank Mechanism by a Mixed System of DAEs and PDEs, Mathematical Modelling of Systems, 2 (1996), 1–18.

[8] Y. Wagner, Modelling and Numerical Simulation of a Heat Exchanger. Z. angew. Math. Mech., 78 (1998), Suppl. 3, 1125-1126.

MODELING OF LUMPED MECHATRONIC SYSTEMS AND CALCULUS OF VARIATIONS

Kurt Schlacher, Werner Haas
Department of Automatic Control
Johannes Kepler University of Linz
Altenbergerstraße 69, A-4040 Linz
kurt[werner]@regpro.mechatronik.uni-linz.ac.at

Abstract. This contribution deals with problems of calculus of variations constraint by a set of differential and algebraic equations (DAEs). DAEs are a natural approach for the modelling of lumped parameter systems in Mechatronics. We present a method for a class of DAEs, which are transformable to an explicit control system in principle. Since the proposed approach does not use this transform explicitly, only a transform to a canonical form for DAEs is used, the Euler-Lagrange equations of the variational problem are derivable by pure algebraic manipulation.

Introduction

In the mathematical modeling of lumped parameter systems in Mechatronics it has turned out that the DAE-system approach [1] is a very natural one. A DAE-system, also called descriptor system, is a set of implicit ordinary differential equations, which are linear in the derivatives, such that the relations

$$\sum\nolimits_{j=1}^{n} e_j^i \dot{x}^j = f^i(t,x,u) \ , \qquad \dot{x}^i = \frac{\mathrm{d}}{\mathrm{d}t} x^i \ , \qquad i = 1,\dots,n \tag{1}$$

are met with a singular matrix $\left[e_j^i\right]$. x denotes the descriptor state and $u \in \mathcal{R}^m$ the descriptor input of the system. Neither x is the usual state of a dynamical system nor u the input of a dynamical system. In [4],[5] it was shown that the descriptor approach becomes easier, if we give up the separation of x and u and merge them to $z = (x,u)$ such that x and u are considered of the same level.

The main goal of the contribution is the geometric approach for the determination of the variational equations with respect to descriptor systems. We start with an objective function $\psi = \int l(t,z)\,\mathrm{d}t$ and a set $\left\{g^i(t,z,\dot{z}) = 0\right\}$ of constraints and look for an integral curve γ such that ψ reaches an extrema. The geometric description starts with the total space $\mathcal{S} = \mathcal{R} \times \mathcal{M}$ with local coordinates $\left(t, z^1, \dots, z^n\right)$, which encompass the independent coordinate t and the dependent coordinates $z = \left(z^1, \dots, z^n\right)$. A natural assumption for a system of differential equations of first order is that the constraints define a submanifold in the first jet bundle $\mathcal{S}^{(1)}$ of the total space $\mathcal{S}$. Since (1) is affine in the derivatives, we get a simpler description, if we consider (1) to be a submodule of the cotangent bundle $\mathcal{T}^*(\mathcal{S})$. This observation leads immediately to the theory of Pfaffian systems defined in the cotangent bundle $\mathcal{T}^*(\mathcal{S})$ of $\mathcal{S}$. Recently it has been shown [4], [5] that the use of a special canonical form for descriptor systems allows to handle the system (1) in a way similar to an explicit control system. Therefore, this form offers the extension of many results known for explicit control systems to descriptor systems. The contribution will prove that this approach has many benefits for the variational problem, too.

This paper is organized as follows. In the first section, the variational problem is stated in detail. In section 2 we derive a Pfaffian system, which describes the necessary conditions of the variational problem and we discuss the properties of the canonical Pfaffian system associated to the descriptor system. This involves the determination of a set of input functions for the descriptor system as well as the derivation of the real state in the sense of an explicit control system. One important result for the variational problem is the simple calculation of a basis for the admissible variations. Finally, an example shows that differential forms are a useful tool to solve this problem.

Problem Statement

To start, we consider the following problem. Let $\mathcal{M}$ denote an n-dimensional smooth manifold with finite n and local coordinates $z = \left(z^1, \dots, z^n\right)$ and let $\gamma(t) = \left(t, \gamma^1(t), \dots, \gamma^n(t)\right) : \mathcal{R} \to \mathcal{R} \times \mathcal{M}$ be any

solution of the system

$$g^i(t,z,\dot{z}) = \sum_{j=1}^{n} e_j^i(t,z)\,\dot{z}^j - f^i(t,z) = 0\,, \qquad \dot{z}^i = \frac{\mathrm{d}}{\mathrm{d}t}z^i\,, \qquad i = 1,\dots,k\,, \qquad k \le n \tag{2}$$

with singular matrix $\left[e_j^i\right]$ of constant rank in the neighborhood of a generic point $z \in \mathcal{M}$. Then, we look for the determining equations of a solution $\gamma(t)$, $t \in [t_1, t_2]$ such that the functional

$$\psi(\gamma) = \int_{t_1}^{t_2} \gamma^*\left(l(t,z)\,\mathrm{d}t\right) = \int_{t_1}^{t_2} l(\gamma)\,\mathrm{d}t \tag{3}$$

is minimized with fixed terminal points $\gamma(t_1), \gamma(t_2)$. Here, γ^* denotes the pullback operation $\gamma^* : \mathcal{T}^*(\mathcal{R}\times\mathcal{M}) \to \mathcal{T}^*\mathcal{R}$ for 1-forms. The system (2) is called a descriptor system and it allows to mix ordinary differential equations with algebraic ones. Let $\mathcal{S} = \mathcal{R}\times\mathcal{M}$ denotes the total space with coordinates $\left(t, z^1, \dots, z^n\right)$ and $\mathcal{S}^{(1)}$ the 1-jet space of $\mathcal{S}$ with standard coordinates $\left(t, z^1, \dots, z^n, \dot{z}^1, \dots, \dot{z}^n\right)$, then the prolongation of γ to the 1-jet space is denoted by $\mathrm{pr}(\gamma) = \left(t, \gamma^1, \dots, \gamma^n, \dot{\gamma}^1, \dots, \dot{\gamma}^n\right)(t)$. Obviously, γ is a solution of (2) iff $g^i(\mathrm{pr}(\gamma)) = 0$ is met. The standard approach to the variational problem above starts with the set of all 1-dimensional point transforms $\varphi_\tau : \mathcal{S} \to \mathcal{S}$ such that the independent coordinate t remains unchanged and $\varphi_0(\gamma) = \gamma$, $\varphi_\tau(\gamma)(t_1) = \gamma(t_1)$, $\varphi_\tau(\gamma)(t_2) = \gamma(t_2)$, $\varphi_{\tau_1+\tau_2} = \varphi_{\tau_1}\circ\varphi_{\tau_2}$ as well as $g^i(\mathrm{pr}(\varphi_\tau(\gamma))) = 0$ is fulfilled. Here, $\mathrm{pr}(\varphi_\tau)$ denotes the prolongation of the point transform φ_τ to the point transform $\mathrm{pr}(\varphi_\tau) : \mathcal{S}^{(1)} \to \mathcal{S}^{(1)}$. Roughly spoken, such an admissible map φ_τ transforms a solution of (2) into another one. In addition, the minimizing curve γ must fulfill the inequality $\psi(\gamma) \le \psi(\varphi_\tau(\gamma))$. Now, it is well known that the determining equations for the extremal solution γ can be derived from the conditions given by

$$\frac{\mathrm{d}}{\mathrm{d}\tau}\, g^i\left(\mathrm{pr}\left(\varphi_\tau(\gamma)\right)\right)\Big|_{\tau=0} = 0\,, \quad i = 1,\dots,k \qquad \text{and} \qquad \frac{\mathrm{d}}{\mathrm{d}\tau}\int_{t_1}^{t_2} l\left(\varphi_\tau(\gamma)\right)\mathrm{d}t\Big|_{\tau=0} = 0\,. \tag{4}$$

To proceed, we need some facts from differential geometry concerning the point transforms φ_τ. The special point transform φ_τ defines a vector field $v \in \mathcal{T}(\mathcal{S})$, $v = \sum_{i=1}^{n} V^i \partial_{z^i}$ by $V^i = \partial_\tau \varphi_\tau^i\big|_{\tau=0}$, which can be prolonged to the 1-jet space by

$$\mathrm{pr}(v) \in \mathcal{T}\left(\mathcal{S}^{(1)}\right)\,, \qquad \mathrm{pr}(v) = \frac{\partial}{\partial\tau}\mathrm{pr}(\varphi_\tau)\Big|_\tau = \sum_{i=1}^{n}\left(V^i\partial_{z^i} + \frac{\mathrm{d}}{\mathrm{d}t}V^i\partial_{\dot{z}^i}\right)\,. \tag{5}$$

Then, a short calculation shows that (4) can we rewritten as

$$L_{\mathrm{pr}(v)}\left(g^i(\mathrm{pr}(\gamma))\right) = 0, \quad i = 1,\dots,k \qquad \text{and} \qquad \int_{t_1}^{t_2} L_v\left(l(\gamma)\,\mathrm{d}t\right) = 0 \tag{6}$$

such that these relations are met for all admissible vector-fields v. Here, L denotes the Lie-derivative of the functions g^i and the 1-form $l\mathrm{d}t$ along the vector-field $\mathrm{pr}(v)$.

Main Results

Equivalently to (2), we consider the Pfaffian system

$$I : \left\{\theta^i\right\}\,, \qquad \theta^i = \omega^i - f^i(t,z)\,\mathrm{d}t\,, \qquad \omega^i = \sum_{j=1}^{n} e_j^i(t,z)\,\mathrm{d}z^j\,, \qquad i = 1,\dots,k \tag{7}$$

with $\theta^i \in \mathcal{T}^*(\mathcal{S})$ and the exterior derivative d of a p-form. A curve γ is a solution of (7), iff $\gamma^*\left(\theta^i\right) = 0$ is met for all 1-forms of (7). In addition, the so called independence condition $\frac{\mathrm{d}}{\mathrm{d}t}\gamma\rfloor\mathrm{d}t > 0$ with the interior product $\rfloor$ of a vector-field and a p-form is satisfied, i.e. the time t is increasing and one can choose t as the independent variable.

One can show [3] that the equations (6) are equivalent to the set

$$v\rfloor\mathrm{d}\left(l\mathrm{d}t + \sum_{i=1}^{k}\lambda^i\theta^i\right) = 0 \mod \mathcal{N}\,, \quad \theta^i \in I\,, \tag{8}$$

which depends only on the field v, but does not depend on its prolongation $\mathrm{pr}\,(v)$ any more. The notation $\sigma = 0 \,\mathrm{mod}\mathcal{N}$ means that the relations must be fulfilled on the submanifold $\gamma([t_1,t_2]) = \mathcal{N} \subset \mathcal{S}$ only. Still, there remains the problem to find a suitable basis for all admissible variational fields v.

First, we discuss this question with respect to the general explicit control system

$$\dot{x}^i = f^i(x,u)\ ,\quad i = 1,\dots,n-m \qquad \text{or} \qquad I: \left\{\theta^i = \mathrm{d}x^i - f^i(x,u)\,\mathrm{d}t\right\} \tag{9}$$

with $z = (x,u)$, $u \in \mathcal{R}^m$ and a given 1-form $l(t,z)\,\mathrm{d}t$. A short calculation shows that $B^* = \left\{\mathrm{d}t, \theta^i, \mathrm{d}u^j\right\}$ is a basis of $\mathcal{T}^*(\mathcal{S})$. The canonical dual basis $B = \{\partial_t, \partial_{\theta^i}, \partial_{u^j}\}$ follows from the relations $v_i \rfloor \omega^j = \delta_j^i$, $v^i \in B$, $\omega_j \in B^*$. The relation

$$\mathrm{d}\phi = \mathrm{d}\left(l\mathrm{d}t + \sum\nolimits_{i=1}^{n-m} \lambda^i\theta^i\right) = \mathrm{d}l \wedge \mathrm{d}t + \sum\nolimits_{i=1}^{n-m}\left(\mathrm{d}\lambda^i \wedge \theta^i + \lambda^i \mathrm{d}\theta^i\right) \tag{10}$$

leads to

$$\begin{aligned} \partial_{\theta^i}\rfloor\mathrm{d}\phi &= (\partial_{\theta^i}\rfloor\mathrm{d}l)\,\mathrm{d}t + \textstyle\sum_{l=1}^{n-m}\lambda^l\left(\partial_{\theta^i}\rfloor\mathrm{d}\theta^l\right) - \mathrm{d}\lambda^i = 0 \ \mathrm{mod}\mathcal{N}\,,\quad i = 1,\dots,n-m\,,\\ \partial_{u^j}\rfloor\mathrm{d}\phi &= (\partial_{u^j}\rfloor\mathrm{d}l)\,\mathrm{d}t + \textstyle\sum_{l=1}^{n-m}\lambda^l\left(\partial_{u^j}\rfloor\mathrm{d}\theta^l\right) = 0 \ \mathrm{mod}\mathcal{N}\,,\quad j = 1,\dots,m\,. \end{aligned} \tag{11}$$

Since we deal with an explicit control system we have $\partial_{\theta^i} = \partial_{x^i}$, $\partial_{x^i}\rfloor\mathrm{d}\theta^l = -\partial_{x^i}f^l\mathrm{d}t$ and $\partial_{u^j}\rfloor\mathrm{d}\theta^l = -\partial_{u^j}f^l\mathrm{d}t$ in addition. Combining these relations, we obtain the well known set

$$\begin{aligned} \partial_{\theta^i}\rfloor\mathrm{d}\phi &= \left(\partial_{\theta^i}\rfloor\mathrm{d}l - \textstyle\sum_{l=1}^{n-m}\lambda^l\partial_{\theta^i}\rfloor\mathrm{d}f^l\right)\mathrm{d}t - \mathrm{d}\lambda^i = 0 \ \mathrm{mod}\mathcal{N}\,,\quad i = 1,\dots,n-m\,,\\ \partial_{u^j}\rfloor\mathrm{d}\phi &= \left(\partial_{u^j}\rfloor\mathrm{d}l - \textstyle\sum_{l=1}^{m}\lambda^l\partial_{u^j}\rfloor\mathrm{d}f^l\right)\mathrm{d}t = 0 \ \mathrm{mod}\mathcal{N}\,,\quad j = 1,\dots,m\,, \end{aligned} \tag{12}$$

It is worth to mention that (12) establishes a descriptor system with the constraints $\partial_{u^j}l = \sum_{l=1}^{n-m}\lambda^l\partial_{u^j}f^l$.

In the case of descriptor systems (7), we have to face the problem that neither the input u is explicitly given nor a suitable basis B^* of $\mathcal{T}^*(\mathcal{S})$ for the derivation of the variational fields v is known. On the other hand, if there exists a basis $B^* = \left\{\mathrm{d}t, \theta^i, \vartheta^j\right\}$ with canonical dual basis $B = \{\partial_t, \partial_{\theta^i}, \partial_{\vartheta^j}\}$, then the derivation of the Euler-Lagrange equations of the variational problem (2), (3) is a straightforward problem. They follow directly from (11) with the substitution $\vartheta^j = \mathrm{d}u^j$ and the result is a complicated DAE-system in general. One can show that under some mild rank conditions the system (7) can be transformed to

$$\tilde{P} = \left(\left\{\tilde{\theta}^i, \mathrm{d}\tilde{g}^j\right\}, \{\tilde{g}^j\}\right)\ ,\qquad \tilde{\theta}^i = \tilde{\omega}^i - \tilde{f}^i\mathrm{d}t\ ,\qquad i = 1,\dots,a\ ,\qquad j = 1,\dots,b \tag{13}$$

such that $\left\{\tilde{\theta}^i, \mathrm{d}\tilde{g}^j\right\}$ describes a Pfaffian system, which is constrained to a submanifold defined by $\tilde{g}^j = 0$ [5]. Iff the integrability conditions

$$\mathrm{d}\tilde{\omega}^i = \sum\nolimits_{i=1}^{a}\alpha_j^i \wedge \tilde{\omega}^j + \sum\nolimits_{i=1}^{b}\beta_j^i \wedge \mathrm{d}\tilde{g}^j \tag{14}$$

are met for suitable 1-forms α_j^i, β_j^i, then one can show that (13) is transformable to the explicit form (12), see [4], [5] for more details. Now we are able to choose functions $\tilde{u}^l$ such that $\bigwedge_{i=1}^{a}\tilde{\omega}^i \wedge \bigwedge_{i=1}^{b}\mathrm{d}\tilde{g}^i\bigwedge_{l=1}^{n-a-b}\mathrm{d}\tilde{u}^l \neq 0$ is fulfilled, then the required basis B^* is given by $B^* = \left\{\mathrm{d}t, \tilde{\theta}^i, \mathrm{d}\tilde{g}^j, \mathrm{d}\tilde{u}^l\right\}$. The canonical dual basis $B = \left\{\partial_t, \partial_{\tilde{\theta}^i}, \partial_{\tilde{g}^j}, \partial_{\tilde{u}^l}\right\}$ follows from the relations $v_i\rfloor\omega^j = \delta_j^i$, $v^i \in B$, $\omega_j \in B^*$. From now on, we assume that the system (7) is transformable to an explicit control system like (12) but we deal with the representation (13), since one can transform (7) to (13) by pure algebraic manipulations, see [2], [4], [5]. Using (14) and the relations $\mathrm{d}\tilde{g}^j = 0 \ \mathrm{mod}\mathcal{N}$, $\tilde{\omega}^i = \tilde{f}^i\mathrm{d}t \ \mathrm{mod}\mathcal{N}$, we can simplify the relation $\phi = \left(l\mathrm{d}t + \sum_{i=1}^{a}\lambda^i\theta^i + \sum_{i=1}^{b}\lambda^{i+a}\mathrm{d}\tilde{g}^j\right)$ and get

$$\begin{aligned} \partial_{\tilde{X}^i}\rfloor\mathrm{d}\phi &= \left(\partial_{\tilde{X}^i}\rfloor\mathrm{d}l + \textstyle\sum_{l=1}^{a}\lambda^l\left(\sum_{h=1}^{a}\left(\partial_{\tilde{X}^i}\rfloor\alpha_h^l\right)f^h - \partial_{\tilde{X}^i}\rfloor\mathrm{d}f^l\right)\right)\mathrm{d}t - \mathrm{d}\lambda^i = 0 \ \mathrm{mod}\mathcal{N}\,,\\ \partial_{\tilde{u}^j}\rfloor\mathrm{d}\phi &= \left(\partial_{\tilde{u}^j}\rfloor\mathrm{d}l + \textstyle\sum_{l=1}^{a}\lambda^l\left(\sum_{h=1}^{a}\left(\partial_{\tilde{u}^j}\rfloor\alpha_h^i\right)f^h - \partial_{\tilde{u}^j}\rfloor\mathrm{d}f^l\right)\right)\mathrm{d}t = 0 \ \mathrm{mod}\mathcal{N}\,, \end{aligned} \tag{15}$$

$i = 1,\dots,a+b$, $j = 1,\dots,n-a-b$ with $\tilde{X}^i = \partial_{\tilde{\theta}^i}$, $\tilde{X}^{i+a} = \partial_{\tilde{g}^j}$ for the basis above. The first set of 1-forms are convertible to ODEs in a straightforward manner, the last set encompasses the constraints. (15) forms the set of the so called Euler-Lagrange-equations of the variational problem (3) and (4). It is worth to compare (12) and (15).

Example

We consider the simple example of a chemical reactor [1]

$$\dot{C} = K_1 (C_0 - C) - R \,, \quad \dot{T} = K_1 (T_0 - T) + K_2 R - K_3 (T - T_C) \,, \quad R - K_3 C e^{-K_4/T} = 0 \,, \tag{16}$$

which describes a first-order isomerization reaction. Here C_0 and T_0 are the known feed reactant concentration and feed temperature. C and T are the corresponding quantities in the product. R is the reaction rate per unit volume, the actuator signal T_C is the temperature of the cooling medium and the K_i, $i = 1,2,3,4$ are constants. The associated Pfaffian system in canonical form is given by

$$\begin{gathered} \tilde{\theta}^1 = \mathrm{d}C - (K_1 (C_0 - C) - R)\,\mathrm{d}t \,, \quad \tilde{\theta}^2 = \mathrm{d}T - (K_1 (T_0 - T) + K_2 R - K_3 (T - T_C))\,\mathrm{d}t, \\ \mathrm{d}g = \mathrm{d}\left(R - K_3 C e^{-K_4/T}\right) \qquad \text{and} \qquad g = R - K_3 C e^{-K_4/T} = 0 \,. \end{gathered} \tag{17}$$

A dual basis B^* of the canonical Pfaffian system is given with $\left(\mathrm{d}t, \tilde{\theta}^1, \tilde{\theta}^2, \mathrm{d}g, \mathrm{d}T_C\right)$ and its canonical dual basis with $B = \left\{\partial_t, \partial_{\tilde{\theta}^1}, \partial_{\tilde{\theta}^2}, \partial_g, \partial_{T_C}\right\}$. Using $\phi = l\,(t, C, T, R, T_C)\,\mathrm{d}t + \lambda^1 \tilde{\theta}^1 + \lambda^2 \tilde{\theta}^2 + \lambda^3 \mathrm{d}g$, we obtain the variational equations

$$\begin{aligned} &\partial_{\tilde{\theta}^1} \rfloor \mathrm{d}\phi = -\mathrm{d}\lambda^1 + \left(\tfrac{\partial}{\partial C} l - K_1 \lambda^1 - \left(-\lambda^1 + K_2 \lambda^2 - \tfrac{\partial}{\partial R} l\right) K_2 e^{-K_4/T}\right) \mathrm{d}t = 0 \mod \mathcal{N} \,, \\ &\partial_{\tilde{\theta}^2} \rfloor \mathrm{d}\phi = -\mathrm{d}\lambda_2 + \left(K_3 \lambda^2 - K_1 \lambda^2 + \tfrac{\partial}{\partial T} l - T^{-2} \left(K_2 \lambda^2 - \lambda^1 - \tfrac{\partial}{\partial R} l\right) K_3 C K_4 e^{-K_4/T}\right) \mathrm{d}t = 0 \mod \mathcal{N} \,, \\ &\partial_g \rfloor \mathrm{d}\phi = -\mathrm{d}\lambda^3 + \left(\lambda^1 - K_2 \lambda^2 + \tfrac{\partial}{\partial R} l\right) \mathrm{d}t = 0 \mod \mathcal{N} \,, \\ &\partial_{T_c} \rfloor \mathrm{d}\phi = \left(\tfrac{\partial}{\partial T_C} l - K_3 \lambda^2\right) \mathrm{d}t = 0 \mod \mathcal{N} \end{aligned} \tag{18}$$

in a straightforward manner.

Conclusion

This contribution has shown that there is no essential difference in the calculus of variations for explicit control systems and descriptor systems, which are transformable to explicit systems in principle. Based on the presented geometric framework using the mathematical language of Pfaffian systems, the variational equations can be determined in a straightforward manner in the neighborhood of generic points. This approach requires the calculation of a canonical Pfaffian system associated to the descriptor system, which offers the identification of the real input and the real state of a control system. Based on this form, we can derive the equations of a variational problem constrained by a descriptor system by pure algebraic manipulations only, which can be done by a computer algebra system. Finally, an example has shown the feasibility of the proposed approach.

References

1. Brenan K.E., Campbell S.L., and Petzold L.R.: Numerical Solution of Initial-Value Problems in Differential Algebraic Equations, SIAM, 1996.
2. Haas W., Schlacher K., Kugi A.: A Software Package for the Analysis of DAE Control Systems, European Control Conference 99, Karlsruhe, 1999.
3. Griffiths P.A.: Exterior Differential Systems and the Calculus of Variations, Birkhäuser Verlag, 1983.
4. Schlacher K., Kugi A., Haas W.: Geometric Control of a Class of Nonlinear Descriptor Systems, NOLCOS, Enschede, 1998.
5. Schlacher K., Haas W., Kugi A.: Ein Vorschlag für eine Normalform von Deskriptorsystemen, ZAMM, Angew. Math. Mech. 79, pp S21 - S24, 1999.

DESCRIPTOR SYSTEMS: PROS AND CONS OF SYSTEM MODELLING BY DIFFERENTIAL-ALGEBRAIC EQUATIONS

P.C. Müller
Safety Control Engineering
University of Wuppertal, D-42097 Wuppertal, Germany
E-mail: mueller@srm.uni-wuppertal.de

Abstract. In recent years the analysis and synthesis of control systems in descriptor form has been established. The general description of dynamical systems by differential-algebraic equations (DAE) is important for many applications in mechanics and mechatronics, in electrical and electronic engineering, and in chemical engineering as well. In this contribution the pros and cons of system modelling by differential-algebraic equations are discussed and an actual state of the art of descriptor systems is presented. Firstly, the advantages of modelling are touched in general and illustrated in detail by Lagrange's equations of first kind, by subsystem modelling and by the statement of the tracking control problem. Secondly, the development of tools for numerical integration is discussed resulting in the comment that today stable and efficient DAE solvers exist and that the simulation of descriptor systems is not a problem any longer. Thirdly, the methods of analyzing and designing descriptor systems are considered. Here, linear and nonlinear systems have to be distinguished. For linear descriptor systems more or less the required methods to solve usual control tasks are available in principal. But actually a related program package for fast and reliable application of these methods is still missed. However, in the near future such a toolbox is expected. Main difficulties arise for nonlinear problems. A few results on stability and optimal control are known only and still a lot of research work has to be effected. In spite of these deficiencies, all over the descriptor system approach is very attractive for modelling and simulation, and will become attractive more and more for analysis and design.

Introduction

The investigation of dynamical systems in mechanical, electrical or chemical engineering usually requires a mathematical modelling of the system behavior. The increasing complexity of these processes lead on the one side to the development of computer programs automatically generating the governing system equations, cf. [27] for multibody systems, or on the other side to an increase of modular subsystem modelling of which the complete model is composed. Usually, this interconnection-oriented modelling describes the dynamic behavior of the single components by differential equations and the coupling of the subsystems by algebraic equations. Allover, the mathematical model is represented by a combined set of differential and algebraic, i.e. by differential-algebraic equations (DAE). In control engineering we speak about singular control systems or descriptor systems [11].

Models of chemical processes, for example, typically consist of differential equations describing the dynamic balances of mass and energy while additionally algebraic equations account for thermodynamic equilibrium relations, steady-state assumptions, empirical correlations, etc. [4, 8, 21]. Also electrical networks can be considered to be composed by subsystems of network elements (like resistors, capacitors, inductors described by different types of equations including differential equations) and by couplings due to Kirchhoff's laws (described by algebraic equations) [7, 13]. In mechanical systems the differential equations usually describe the dynamics of the subsystems and the algebraic equations characterize couplings by constraints such as joints. A general approach to handle mechanical systems as an interconnected set of dynamic modules has been given in [25]. In the following three examples of descriptor modelling are shortly dealt with for illustration.

Langrange's equations of first kind. Langrange's equations of first and second kind are well established in analytical mechanics, cf. [24]. They describe the dynamic behavior of discrete systems, particularly of multibody systems. The difference of the two kinds consists in the manipulation of the kinematic constraints. If a kinematic description of the system has been performed by generalized coordinates consistent with the constraints the Lagrange's equation of second kind can be applied leading to a set of differential equations only. But if a redundant set of coordinates is used to describe kinematically the system regarding still some constraints explicitly then Lagrange's equations of first kind hold. In case of holonomic constraints

$$f(q) = 0 \tag{1}$$

we have

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}}\right) - \frac{\partial L}{\partial q} = Q + F^T \lambda \tag{2}$$

where the Lagrangian funktion L = T-U consists of the kinetic and potential energies T and U, Q represents the nonconservative forces acting on the system, $\mathbf{F} = \mathbf{F}(q) = \frac{\partial f}{\partial q^T}$ is the Jacobian matrix of the constraints and λ is the vector of Lagrange's multipliers. They represent the constraint forces if the column vectors of $\mathbf{F}^T$ are normalized. While the variables q describe the motion of the system, the Lagrange's multipliers λ give some information on the load of the mechanical structure. Therefore, critical loads due to the motion may be considered simultaneously. Equations (1, 2) represent a system of differential-algebraic equations. If Q includes some actuator forces to control the multibody system then a descriptor system is under consideration.

Subsystem Modelling. If the interconnection-oriented modelling approach is applied [15], usually the dynamics of N subsystems are described by sets of differential equations

$$\dot{\mathbf{x}}_i = \mathbf{a}_i(\mathbf{x}_i, \mathbf{u}_i),\ i = 1, .., N, \tag{3}$$

where $\mathbf{x}_i$ are the internal state vectors and $\mathbf{u}_i$ the control vectors of the corresponding subsystems. The couplings among the subsystems may be obtained kinematically by "constraints" or kinetically by "forces" leading to

$$\dot{\mathbf{x}}_i = \mathbf{a}_i(\mathbf{x}_i, \mathbf{u}_i) + \sum_{j=1}^{N} \mathbf{a}_{ij}(\mathbf{x}_i, \mathbf{x}_j) + \sum_{j=1}^{N} \mathbf{L}_{ij}(\mathbf{x}_j)\lambda_j \tag{4}$$

$$0 = \sum_{j=1}^{N} \mathbf{f}_{ij}(\mathbf{x}_j),\ i = 1, .., N. \tag{5}$$

The additional terms compared to (3) are the kinetic couplings $\mathbf{a}_{ij}$ between subsystems no. i and j and the kinematic couplings (5) which have to be considered in the dynamic balance equations (4) by some Lagrange's multipliers λ_j with some input matrices $\mathbf{L}_{ij}$ due to the coupling requirements. How $\mathbf{L}_{ij}$ is defined more precisely depends on the physical principles behind the system discipline; equations (1, 2) show an example of mechanical systems. All over, equations (4, 5) represent again a descriptor system.

Tracking control. In control engineering often the problem of tracking control arises, e.g. the prescribed path control of a robot. In this case the process dynamics may be described in the state space by

$$\dot{\mathbf{x}} = \mathbf{a}(\mathbf{x}, \mathbf{u}, t) \tag{6}$$

and it is asked for the control u which guarantees that some output variables y = c (x, u, t) follow a prescribed reference path $y_{ref}(t)$:

$$0 = \mathbf{c}(\mathbf{x}, \mathbf{u}, t) - y_{ref}(t). \tag{7}$$

This descriptor system (6, 7) can be described very smart in the case of dim u = dim y. Then the descriptor system

$$\begin{bmatrix} \mathbf{I}_x & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\overline{\mathbf{x}}} \end{bmatrix} = \begin{bmatrix} \mathbf{a}(\mathbf{x}, \overline{\mathbf{x}}, t) \\ \mathbf{c}(\mathbf{x}, \overline{\mathbf{x}}, t) - y_{ref}(t) \end{bmatrix} \tag{8}$$

defines explicitly the desired tracking control

$$\mathbf{u}(t) = \begin{bmatrix} 0 & \mathbf{I}_u \end{bmatrix} \begin{bmatrix} \mathbf{x}(t) \\ \overline{\mathbf{x}}(t) \end{bmatrix} \tag{9}$$

where $\mathbf{I}_x$, $\mathbf{I}_u$ are identity matrices of dim(x) and dim(u) respectively.

Pros and cons. With respect to the tasks of system modelling the descriptor system approach has many advantages. It is a very natural way to model process dynamics. It refers much more to the physical behavior of the system and gives more physical insight. The interpretation of results is also more simple than in case of the more abstract description by state space models. In the opposite the state space system approach was mainly required by the mathematic tools available until 1980 to simulate, to analyze and to design such systems.

Simulation

As long as it was not possible to simulate descriptor systems very efficient and very accurate still the state space approach was superior according to the well established tools of numerical integration of ordinary differential equations. But in the '70s the simultaneous numerical solution of differential and algebraic equations was firstly considered [5]. Step by step numerical system solvers were developed. For index-1-problems (see below) the code DASSL has been presented [22], stimulating more research also for higher index problems. In the meantime a lot of efficient solvers for DAE's have been developed, cf. [2, 6, 28]. In a more recent Ph.D. thesis [26] on the modular simulation of mechatronic systems several solvers have been compared resulting in the recommendation of the codes SDOP853 and SDOPRI5 which are modified versions of Runga-Kutta solvers for ordinary differential equations including projection steps with respect to the constraints of the algebraic equations. With respect to

these results today a number of stable and efficient DAE solvers exist and can be applied as naturally as ODE solvers for state space models.

Analysis and Synthesis

The tools for the analysis and synthesis of descriptor systems have been developed enormously in the last two decades. As usual, linear theory has been in the foreground of the discussion, but first results on nonlinear problems have been reported, too.

Linear time-invariant descriptor systems are presented by

$$\mathbf{E}\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \tag{10}$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \tag{11}$$

where $\mathbf{x}$ is an n-dimensional descriptor vector, $\mathbf{u}$ denotes the r-dimensional control input vector, and $\mathbf{y}$ characterizes the m-dimensional measurement output vector. The matrices $\mathbf{E}$, $\mathbf{A}$ are n x n-matrices, and $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$ have dimensions n x r, m x n, m x r, respectively. The essential property of descriptor systems is that $\mathbf{E}$ is a singular matrix

$$\text{rank } \mathbf{E} < n, \tag{12}$$

such that (10) consists of differential and algebraic equations.

The basic tool in discussing (10) is the theory of the matrix pencil ($s\mathbf{E} - \mathbf{A}$) by Weierstrass and Kronecker in the last century, cf. [3], separating the system into a few subsystems with different properties. Assuming unique behavior of (10) for all control inputs, i.e. assuming that the matrix pencil is regular,

$$p(s) \equiv \det(s\mathbf{E} - \mathbf{A}) \not\equiv 0, \tag{13}$$

then system (10) is strictly equivalent to the Weierstrass-Kronecker form

$$\dot{\mathbf{x}}_1(t) = \mathbf{A}_1\mathbf{x}_1(t) + \mathbf{B}_1\mathbf{u}(t), \tag{14}$$

$$\mathbf{N}_k\dot{\mathbf{x}}_2(t) = \mathbf{x}_2(t) + \mathbf{B}_2\mathbf{u}(t), \tag{15}$$

$$\mathbf{y}(t) = \mathbf{C}_1\mathbf{x}_1(t) + \mathbf{C}_2\mathbf{x}_2(t). \tag{16}$$

Equation (14) represents the "slow subsystem" of dimension n_1, and the n_2-dimensional "fast subsystem" is described by (15). The n_2 x n_2-matrix $\mathbf{N}_k$ is nilpotent of degree k ($\mathbf{N}_k^{k-1} \neq \mathbf{0}, \mathbf{N}_k^k = \mathbf{0}$) defining the index k of the linear descriptor system.

According to the separation into the two subsystems controllability and observability investigations split off into at least two different concepts of so-called R/I-controllability and –observability guaranteeing different properties of a feedback control, cf. [3 , 10]. The results of many investigations in the 80's have been summarized by Lewis [10] and Dai [3]. Stability can be discussed by the eigenvalues of the matrix pencil ($s\mathbf{E} - \mathbf{A}$), i.e. by the roots of the characteristic polynomial (13). Another approach is based on the generalized matrix equation

$$\mathbf{A}^T\mathbf{P}\mathbf{E} + \mathbf{E}^T\mathbf{P}\mathbf{A} = -\mathbf{Q} \tag{17}$$

where definiteness properties of $\mathbf{P}$ and $\mathbf{Q}$ with respect to certain subspaces assure stability [14]. First results on the design of linear feedback control by pole placement have been presented in [3]. But the main problem of the synthesis of feedback control consists in the possibility of non-proper system behavior. This can be seen immediately by the solution of the fast subsystem (15), cf. [3],

$$\mathbf{x}_2(t) = -\mathbf{B}_2\mathbf{u}(t) - \mathbf{N}_k\mathbf{B}_2\dot{\mathbf{u}}(t) - \ldots - \mathbf{N}_k^{j-1}\mathbf{B}_2\mathbf{u}^{(j-1)}(t), \tag{18}$$

which includes generally higher-order time-derivatives of the control input. The two cases have to be distinguished where the solution of (10) (or (14 and (15)) depends either only on $\mathbf{u}(t)$ but not on its derivatives $\dot{\mathbf{u}}(t),\ldots,\mathbf{u}^{(j-1)}(t)$ or on $\mathbf{u}(t)$ and its derivatives $\dot{\mathbf{u}}(t),\ldots,\mathbf{u}^{(j-1)}(t)$ according to the general case (18). In the first case the system is called "proper", in the second case "non-proper" according to related proper and non-proper transfer matrix functions. The system (10) is proper if and only if in the representation (14, 15) the equation

$$\mathbf{N}_k\mathbf{B}_2 = 0 \tag{19}$$

holds. The distinction between proper and non-proper descriptor systems and its consequences for the control design has been discussed just recently [18, 19, 20]. Regarding proper and non-proper systems in different ways, the linear quadratic optimal regulator problem [18, 19] and the descriptor state estimation problem [20] has been discussed in detail and properly solved. Therefore, the standard design methods are available for linear descriptor systems, too.

Also first results on robust control design exist. The H_∞-control problem has been considered in [9, 12, 23, 29]. But sometimes some assumptions such as I-controllability have been introduced to solve the problem. Therefore, research work is still necessary to loosen such conditions for more general descriptor systems. For example,

a linear descriptor system (1, 2) is never I-controllable but it may be R-controllable and a H_∞-control design is still of interest.

The analysis and the design of nonlinear descriptor systems is still an open field. Essential results exist with respect to the stability problem only [1, 16, 17]. Some steps have been taken for the optimal control design [17], but especially for non-proper systems many problems still have to be solved. The method of exact linearization and nonlinear decoupling by state feedback has to be generalized to descriptor systems. The problem is under consideration.

Pros and cons. For the analysis and synthesis of linear descriptor systems the usual theoretical tools are available (more or less). Related program packages are being developed. Usually they are based on van Dooren's algorithm characterizing the eigenstructure problem of the matrix pencil (sE - A) [30]. Therefore some research groups are partly provided with computer algorithms for the analysis and design of linear descriptor systems, but still we do not have the standard or the comfort of a MATLAB toolbox. But it is expected that this is only a matter of time. In a few years the linear descriptor toolbox will be available as linear state space algorithms today.

For nonlinear descriptor systems still a lot of research work has to be done. Here, we are only on the beginning to understand nonlinear system behavior and to develop control design methods.

Conclusions

In this contribution an effort has been made to characterize the state of the art of modelling, analyzing and designing dynamical processes by the descriptor system approach. Without any doubts the modelling of dynamical systems by differential-algebraic equations has many advantages and is superior to state space modelling. The simulation tools for DAE's are well established and are comparable with ODE solvers. The weak points of the descriptor approach are still considered to be connected with the analysis and the design of such systems. But for linear descriptor systems it is expected that a related MATLAB toolbox will be available within a short time. For nonlinear descriptor systems a lot of research work has still to be performed.

References

1. Bajic, V., Lyapunov's Direct Method in the Analysis of Singular Systems and Networks. Shades Technical Publications, Hillcrest, RSA, 1992.
2. Brenan, K.E., Campbell, S.L. and Petzold, L.R., Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations. North-Holland, New York, 1989.
3. Dai, L., Singular Control Systems. Lecture Notes in Control and Information Sciences, Vol. 118, Springer, Berlin-Heidelberg, 1989.
4. Eich, E., Burr, P., Kröner, A. and Lory, P., Modellierung und numerische Simulation in der chemischen Verfahrenstechnik. In: Mathematik in der Praxis (Eds.: Bachem, A., Jünger, M. and Schrader, R.) Springer, Berlin-Heidelberg, 1995, 61-85.
5. Gear, C.W., The Simultaneous Numerical Solution of Differential-Algebraic Equations. IEEE Trans. Circuit Theory, 18 (1971), 89-95.
6. Hairer, E. and Wanner, G., Solving Ordinary Differential Equations II, Stiff and Differential-Algebraic Problems. Springer, Berlin-Heidelberg, 1991.
7. Kampowsky, W., Rentrop, P. and Schmidt, W., Classification and Numerical Simulation of Electric Circuits. Surveys on Mathematics for Industry, 2 (1992), 23-65.
8. Kumar, A. and Daoutidis, P., Feedback Control of Nonlinear Differential-Algebraic-Equation Systems. AIChE Journal, 41 (1995), 619-636.
9. Kwakernaak, H., Frequency Domain Solution of the H_∞-Problem for Descriptor Systems. In: Learning, Control and Hybrid Systems (Eds.: Yamamoto,Y. and Hara, S.), Springer, London, 1999.
10. Lewis, F.L., A Survey of Linear Singular Systems. Circuits, Systems and Signal Processing, 5 (1986), 3-36.
11. Luenberger, D.G., Dynamic Equations in Descriptor Form. IEEE Trans. Automatic Control, 22 (1977), 312-321.
12. Masubuchi, I., Kamitane, Y. Ohara, A. and Suda, N., H_∞-Control for Descriptor Systems: A Matrix Inequalities Approach. Automatica, 33 (1997), 669-673.
13. Mathis, W., Analysis of Linear Time-Invariant Networks in the Frequency Domain. In: Mathematical Modelling and Simulation of Electrical Circuits and Semiconductor Devices (Eds.: Bank, R.E., Bulirsch, R., Gajewski, H. and Mertens, K.), Birkhäuser, Basel, 1992, 83-90.

14. Müller, P.C., Stability of Linear Mechanical Systems with Holonomic Constraints. Applied Mechanics Review, 46 (1993), No. 11, Part 2, S160-S164.
15. Müller, P.C., Descriptor Systems: A New Way to Model Mechatronic System? In: Proc. 3rd European Control Conference, Rome, 1995, Vol. 3, Part 2, 2725-2729.
16. Müller, P.C., Stability of Nonlinear Descriptor Systems. Z. Angew. Math. Mech., 76 (1996), Supplement 4, 9-12.
17. Müller, P.C., Stability and Optimal Control of Nonlinear Descriptor Systems: A Survey. Appl. Math. and Comp. Sci., 8 (1998), 269-286.
18. Müller, P.C., Analysis and Control Design of Linear Descriptor Systems. In: Advances in Systems, Signals, Control and Computers, Vol. I (Ed.: Bajic, V.), Center for Engineering Research, Technikon Natal, Durban, RSA, 1998, 11-17.
19. Müller, P.C., Linear Control Design of Linear Descriptor Systems. In: Proc. 14th IFAC World Congress, Beijing, 1999, Pergamon, 1999, Vol. C, 31-36.
20. Müller, P.C., Verallgemeinerte Luenberger-Beobachter für lineare Deskriptorsysteme. Z. Angew. Math. Mech., 79 (1999), Supplement 1, S9-S12.
21. Panreck, K., Jahnich, M. and Dörrscheidt, F., Verwendung differential-algebraischer Gleichungen zur verkopplungsorientierten Modellierung komplexer Prozesse. Automatisierungstechnik, 42 (1994), 239-247.
22. Petzold, L.R., A Description of DASSL: A Differential/Algebraic System Solver. In: Scientific Computing (Eds.: Stepleman, R.S. et al.), North-Holland, Amsterdam. 1983, 65-68.
23. Rehm, A. and Allgöwer, F., H_∞-Control of Differential-Algebraic-Equation Systems. To appear.
24. Rosenberg, R.M., Analytical Dynamics of Discrete Systems. Plenum Press, New York 1977.
25. Rükgauer, A. and Schiehlen, W., Simulation of Modular Dynamic Systems. In: Proc. 2nd MATHMOD, Vienna, 1997, 329-334.
26. Rükgauer, A., Modulare Simulation mechatronischer Systeme mit Anwendung in der Fahrzeugdynamik. VDI-Fortschr.-Ber., Reihe 20, Nr. 248, VDI, Düsseldorf, 1997.
27. Schiehlen, W., Multibody Systems Handbook. Springer, Berlin-Heidelberg, 1990.
28. Simeon, B., Numerische Integration mechanischer Mehrkörpersysteme: Projizierende Deskriptorformen, Algorithmen und Rechenprogramme. VDI-Fortschr.-Ber., Reihe 20, Nr. 130, VDI, Düsseldorf, 1994.
29. Takaba, K., Morihira, N., Katayama, T., H_∞-Control for Descriptor Systems - A J-Spectral Factorization Approach. In: Proc. 33rd IEEE Conf. Decision and Control, 1994, 2251-2256.
30. Van Dooren, P., The Generalized Eigenstructure Problem in Linear System Theory. IEEE Trans. Automatic Control, 26 (1981), 111-129.

SEMIDISCRETIZATION MAY ACT LIKE A DEREGULARIZATION

Michael Günther
Technische Universität (TH) Karlsruhe
Institut für Wissenschaftliches Rechnen und Mathematische Modellbildung
Engesserstr.6, D-76128 Karlsruhe

Abstract. In electrical circuit simulation, a refined network approach is used to describe secondary and parasitic effects. This ansatz yields initial-boundary value problems of mixed partial-differential and differential-algebraic equations, so-called PDAE systems. One requirement for method-of-lines applications is that the analytical properties of the approximative DAE system are consistent with the original PDAE system. Especially, both should show the same sensitivity w.r.t. initial and boundary data. It is already known in the literature that semidiscretization may act like a regularization, i.e. the ADAE system is less sensitive than the PDAE model. Considering PDAE network models for interconnected electrical circuits, we show that the opposite can be true: semidiscretization may act like a deregularization of a PDAE network model, if the method-of-lines approach is not consistent with the information flow along characteristics.

1 Introduction

In network simulation packages, real circuit elements and interconnections are commonly replaced by companion models of ideal and compact network elements, whose properties are determined uniquely by fixing electrical parameters like capacitances or inductances. This yields a unique modelling approach, which allows for including parasitic and second order effects into the differential-algebraic (DAE) network approach. Examples are transistor models which approximate the physical behaviour of semiconductor devices by companion models of different modelling levels, or transmission line models, which consist of RLC elements and controlled sources. Mathematically, this approach corresponds to a spatial discretization of the governing partial differential equations (PDEs) already at the modeling level. Another short-coming is the frequent use of arbitrary coupled sources that may destroy the structure of the network equations and thus lead to high-index systems [5, 9].

As an alternative, the co-simulation approach makes use of already existing simulation software for single parts of the system: different parts of the systems are modelled independently of each other and simulated by two simulation packages for electrical networks and electromagnetic fields; coupling is ensured by coupling the simulators. In addition to convergence problems, difficulties may arise, since coupled systems often are characterized by very different time constants.

A third approach is the use of generalized network models [6]. Refined models are allowed for interconnects and semiconductor devices, whose characteristic equations define PDE models. Hence numerical methods can be tailored exactly to the resulting mathematical models—the spatial discretization is not yet made at modeling level. Mathematically spoken, this approach leads to a coupled system of DAEs and PDEs, with the boundary conditions for the PDEs linked to the DAEs at the boundary nodes. Such systems are called partial differential-algebraic equations, shortly PDAE systems.

The analysis of PDAE systems and their numerical discretization is in the focus of actual research: one aim is to generalize the DAE index concept to PDAE systems to get some knowledge on structural properties before discretization [3, 7, 8, 10]: for example, the sensitivity of the solution to small perturbations in the initial data and/or input signals. On the other hand, estimates are required for the impact of semidiscretization on the index of the resulting approximative DAE (ADAE) system: does the ADAE system properly reflect the behaviour of the original PDAE system? Or does one detect an artificial smoothing effect? Or even a coarsening one?

It is already known that the last but one question has to be answered in the affirmative. Semidiscretization may act like a regularization [1, 2]: the ADAE system is less sensitive w.r.t. input data than the PDAE model, and may yield physically incorrect solutions. In this contribution we will affirm the last question, too.

The paper is organized as follows. Introducing generalized network models for interconnects, we will derive PDAE network equations for interconnected electrical networks in the next section. Estimates are given in Sect. 3 for the solution of these mixed systems of DAEs and PDEs in terms of initial

and boundary data. These sensitivity properties should be reflected by the approximative DAE systems. We will investigate for a simple benchmark, the interconnected VC loop, whether this demand on approximative DAE systems is fulfilled or not. We will learn that semidiscretization may act like a deregularization of a regularized model, if the semidiscretization is not consistent with the information flow along characteristics.

2 PDAE network equations for interconnected electrical networks

In the following, we consider two electrical networks which are coupled by a system of d uniform lossy transmission lines. To derive a mathematical model, we use a hybrid network approach: the electrical circuits are described by DAE models, whereas the transmission lines shown in Fig. 1 are governed by a PDE model. Both models are linked via boundary node voltages and currents.

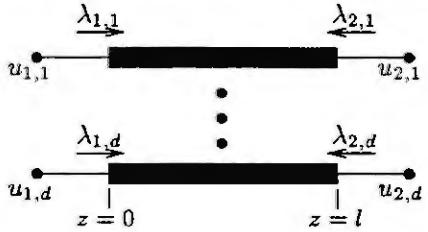


Figure 1: PDE-network model for system of d uniform lossy transmission lines

PDE network model for transmission lines. Assuming quasi stationary behaviour transverse to the wave propagation, the signal propagation in the transmission lines can be characterized by the telegrapher's equations

$$-\boldsymbol{V}_z(z,t) = \boldsymbol{L}\boldsymbol{J}_t(z,t) + \boldsymbol{R}\boldsymbol{J}(z,t), \tag{1a}$$

$$-\boldsymbol{J}_z(z,t) = \boldsymbol{C}\boldsymbol{V}_t(z,t) + \boldsymbol{G}\boldsymbol{V}(z,t), \tag{1b}$$

where $\boldsymbol{R}, \boldsymbol{L}, \boldsymbol{G}$ and $\boldsymbol{C} \in \mathbf{R}^{d\times d}$ are the positive-definite symmetric resistance, inductance, conductance and capacitance matrices per unit length. $\boldsymbol{V}(z,t)$ is a d-dimensional vector of line voltages with respect to ground, and $\boldsymbol{J}(z,t)$ is an d-dimensional vector of line currents. This first order hyperbolic system of partial differential equations is initialized by a set of initial values

$$\boldsymbol{V}(z,t_0) = \boldsymbol{V}^0(z), \tag{2a}$$

$$\boldsymbol{J}(z,t_0) = \boldsymbol{J}^0(z), \tag{2b}$$

$\forall z \in I := [0,l]$ at initial time point t_0.

After introducing d virtual current sources $\lambda_1 := (\lambda_{1,1},\ldots,\lambda_{1,d})$ and $\lambda_2 := (\lambda_{2,1},\ldots,\lambda_{2,d})$ at the boundaries, the characteristic equation for the PDE model of a lossy transmission line system reads

$$\lambda = \begin{pmatrix} \boldsymbol{J}(0,t) \\ -\boldsymbol{J}(l,t) \end{pmatrix} \quad \text{with} \quad \lambda = (\lambda_1, \lambda_2), \tag{3}$$

with the line currents $\boldsymbol{J}$ defined by the telegrapher's equations (1). The PDE model is completed by the boundary conditions

$$\begin{pmatrix} \boldsymbol{V}(0,t) \\ \boldsymbol{V}(l,t) \end{pmatrix} = \begin{pmatrix} \boldsymbol{u}_1 \\ \boldsymbol{u}_2 \end{pmatrix},$$

which couples the PDE network model with the DAE model for both electrical networks.

DAE network model for linear electrical circuits. A network model is used to describe the electrical behaviour of the circuit: network equations for node potentials are derived using Kirchhoff's laws and characteristic equations for the elements. This results in a system of differential-algebraic equations since only topology and no spatial dimension is considered. Using classical Modified Nodal Analysis (MNA), only node potential $\boldsymbol{u}$, currents through inductive and resistive branches $\boldsymbol{\jmath}_L$ and $\boldsymbol{\jmath}_V$, and currents λ

at the boundaries of interconnects are unknowns. The DAE network equations in $x := (y, \lambda)$ with $y := (u, \jmath_L, \jmath_V)$ read

$$\begin{pmatrix} A_C\tilde{C}A_C^\top & 0 & 0 & 0 \\ 0 & \tilde{L} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \dot{x} + \begin{pmatrix} A_R\tilde{G}A_R^\top & A_L & A_V & A_{IC} \\ -A_L^\top & 0 & 0 & 0 \\ -A_V^\top & 0 & 0 & 0 \\ -A_{IC}^\top & 0 & 0 & 0 \end{pmatrix} x + \begin{pmatrix} A_I\imath(t) \\ 0 \\ v(t) \\ \begin{pmatrix} V(0,t) \\ V(l,t) \end{pmatrix} \end{pmatrix} = 0 \qquad (4)$$

with consistent initial values

$$x(t_0) = x_0. \qquad (5)$$

The element-related incidence matrices A_C, A_L, A_R, A_V, A_I and A_{IC} describe the branch-current relations for capacitive, inductive, resistive branches and branches for voltage sources, current sources and transmission line elements. The capacitance, inductance and conductance matrices $\tilde{C}$, $\tilde{L}$ and $\tilde{G}$ are assumed to be positive definite and symmetric [9].

Altogether, (1–5) define a mixed initial-boundary value problem of PDEs and DAEs.

3 Sensitivity analysis for linear networks

To derive estimates for the sensitivity of linear PDAE network equations, we consider the inner product of (1a) and (1b) with J and V, resp. Integration over $(0, l)$ and integration by parts yields

$$\{V^\top(l,t)J(l,t) - V^\top(0,t)J(0,t)\} + \frac{1}{2}\frac{d}{dt}\Big[(LJ,J) + (CV,V)\Big] + (RJ,J) + (GV,V) = 0$$

with $(\cdot,\cdot)$ denoting the inner product in $L_2(I)$. Using the coupling condition in (4) and $A_{IC}^\top A_{IC} = I$, the first term can be replaced by

$$u^\top A_C\tilde{C}A_C^\top\dot{u} + \jmath_L{}^\top\tilde{L}\jmath_L + u^\top A_R\tilde{G}A_R^\top u + u^\top A_I\imath(t) + v(t)^\top\jmath_V.$$

With the symmetry of $\tilde{C}, \tilde{L}$ and C, L we obtain

$$\frac{1}{2}\Big(u^\top A_C\tilde{C}A_C^\top u + \jmath_L{}^\top\tilde{L}\jmath_L + (LJ,J) + (CV,V)\Big)\Big|_0^t + \int_0^t \Big(u^\top\Big(A_R\tilde{G}A_R^\top u + A_I\imath(t)\Big) + v(t)^\top\jmath_V + (RJ,J) + (GV,V)\Big)\,d\tau = 0.$$

In the following we consider different network topologies [5, 9]:

1. *Index-0 case:* $y = (u, \jmath_L)$ *with* $\ker A_C^\top = \{0\}$.
   In this case, we obtain together with the positive definiteness of C and L the estimate

   $$\rho(t) \le \text{const}\,\left(\rho(0) + \int_0^t \left(\|\imath(\tau)\|_2^2 + \rho(\tau)\right)\,d\tau\right) \quad \text{with} \quad \rho(t) := \|V(t)\|_{L_2}^2 + \|J(t)\|_{L_2}^2 + \|y(t)\|_{L_2}^2.$$

   Applying Gronwall's lemma, we end up with

   $$\rho(t) \le \text{const}\,\left(\rho(0) + \int_0^t \|\imath(\tau)\|_2^2\,d\tau\right)\exp(\text{const}\,t) \qquad (6a)$$

   $$\lambda = -A_{IC}^\top\Big[A_C\tilde{C}A_C^\top\dot{u} + A_R\tilde{G}A_R^\top u + A_L\jmath_L + A_I\imath(t)\Big] \qquad (6b)$$

2. *Index-1 case:* $y = (u, \jmath_L)$ *with* $\ker A_C^\top \neq \{0\}$ *and* $\ker(A_CA_R)^\top = \{0\}$.
   Now, one gets an estimate of type (6) with

   $$\tilde{\rho}(t) := \|V(t)\|_{L^2}^2 + \|J(t)\|_{L^2}^2 + \|\jmath_L(t)\|_2^2 + u^\top(t)A_C\tilde{C}A_C^\top u(t) + \int_0^t u^\top(\tau)A_R\tilde{G}A_R^\top u(\tau)\,d\tau$$

   replacing $\rho(t)$.

   In both cases there is no dependence in the estimates for J, V and y on time derivatives — the solution is not sensitive w.r.t. initial values $\rho(0)$ and input signals $\imath$.

3. *Possibly higher-index case:* $y = (u, \jmath_L, \jmath_V)$ *with* $\ker(A_C A_R A_V)^\top = \{0\}$.
In this case, we only get an estimate of the type

$$\bar{\rho}(t) \leq \text{const} \left(\bar{\rho}(0) + \max_{0 \leq \tau \leq t} \left(\|\imath(\tau)\|_2^2 + \|v(\tau)\|_2^2 + \|\jmath_V(\tau)\|_2^2 \right) \right) \exp(\text{const}\, t) \tag{7a}$$

$$\lambda = -A_{IC}^\top \left[A_C \widetilde{C} A_C^\top \dot{u} + A_R \widetilde{G} A_R^\top u + A_V \jmath_V + A_L \jmath_L + A_I \imath(t) \right] \tag{7b}$$

with $\bar{\rho} = \rho$ for $\ker(A_C A_R A_V)^\top = \ker(A_C A_V)^\top = \{0\}$ and $\bar{\rho} = \tilde{\rho}$ otherwise. In this case, the estimates do not exclude the solution to be sensitive w.r.t. initial values $\rho(0)$ and input signals $\imath$ and v.

4 Semidiscretization may deregularize a regularized model

In this section we investigate the sensitivity of a benchmark example w.r.t. initial values and input signals, both for the PDAE network equations and different ADAE systems after semidiscretization w.r.t. space. We use the loop of independent voltage source $v(t)$ and linear capacitance $\widetilde{C}$ shown in Fig. 2, which is connected by a single lossy transmission line.

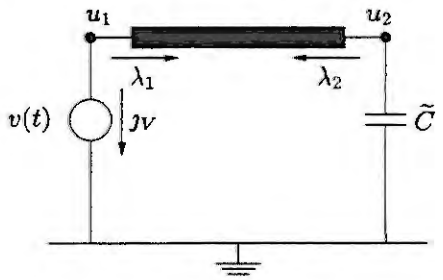


Figure 2: VC loop connected by single transmission line

In this case (4) reads

$$\left(\begin{array}{c|c|c} \binom{0}{1}\widetilde{C}\,(0\ 1) & 0 & 0 \\ \hline 0 & 0 & 0 \\ \hline 0 & 0 & 0 \end{array}\right) \begin{pmatrix} \dot{u} \\ \dot{\jmath}_V \\ \dot{\lambda} \end{pmatrix} + \left(\begin{array}{c|c|c} 0 & \binom{1}{0} & I_2 \\ \hline (1\ 0) & 0 & 0 \\ \hline I_2 & 0 & 0 \end{array}\right) \begin{pmatrix} u \\ \jmath_V \\ \lambda \end{pmatrix} + \begin{pmatrix} 0 \\ -v(t) \\ -\begin{pmatrix} V(0,t) \\ V(l,t) \end{pmatrix} \end{pmatrix} = 0 \tag{8}$$

If the transmission line is skipped, we know from network theory that the DAE network equations have index two: the current $\jmath_V$ through the voltage source depends on the time derivative of the voltage source, i.e. $\jmath_V = -\lambda_1 = -\widetilde{C}\dot{u}$ with $u := u_1 = u_2$ in this case [5].

Since $\ker(A_C A_R A_V)^\top = \ker(A_C A_V)^\top = \ker I_2 = \{0\}$ holds, the PDAE network equations (8) are of type three: the estimates of Sect. 3 do not exclude the solution to be sensitive w.r.t. initial values and input signals. However, this is not true for our example. Neglecting the anyway regularizing source term in (1), line resistance R and conductance G, the PDAE network equations can be solved analytically [7]. The solution V, J, u, $\jmath_V$ and λ do only depend on the initial values for V, J and the time-dependent voltage source $v(t)$, but not on its derivatives. Hence the consideration of transmission lines effects regularizes the DAE network model.

Using semidiscretization w.r.t. space, the method-of-lines approach converts the PDAE network equations into an approximative DAE system in time only. To reflect the physical characteristics of the original PDAE network model, the ADAE system should be neither more nor less sensitive, though the latter might facilitate the numerical solution of the approximative DAE system; however, the obtained solution could be physically incorrect.

If we apply a method-of-lines approach to the telegrapher's equations (1) rewritten in characteristic form, it is easy to show that all standard schemes—central differences, upwind schemes and finite elements—yield ADAE system with index one, in accordance with the properties of the PDAE network equations [7].

The situation is quite different if semidiscretization is applied to the original formulation (1), neglecting the information flow. Central differences now yield index two for some exceptional values of

the discretization parameter h, but index one is assured for h being small enough [7]. This behaviour is well-known for ADAE systems arising from method-of-lines applied to PDAEs [1, 2].

For linear finite elements the approximate DAE system will have index two *for all* discretization parameters and $\boldsymbol{C}, \boldsymbol{L} > 0$ and $\boldsymbol{R}, \boldsymbol{G} \geq 0$ arbitrarily. Time and space are separated by the standard Ritz ansatz

$$\begin{aligned} V(z,t) &= \psi_0(z)u_1(t) + \boldsymbol{\psi}^\top(z)\,\boldsymbol{p}(t) && \text{(left boundary condition)} \\ J(z,t) &= \varphi_0(z)\lambda_1(t) + \boldsymbol{\varphi}^\top(z)\,\boldsymbol{q}(t) && \text{(left boundary condition)} \\ u_2(t) &= \psi_0(1)u_1(t) + \boldsymbol{\psi}^\top(1)\,\boldsymbol{p}(t) && \text{(right boundary condition)} \quad (9a) \\ \lambda_2(t) &= \varphi_0(1)\lambda_1(t) + \boldsymbol{\varphi}^\top(1)\,\boldsymbol{q}(t) && \text{(right boundary condition)} \quad (9b) \end{aligned}$$

with linear finite elements $\boldsymbol{\psi} := (\psi_1, \ldots, \psi_N)$, $\boldsymbol{\varphi} := (\varphi_1, \ldots, \varphi_N)$ as ansatz functions and time dependent coefficients $\boldsymbol{p} := (p_1, p_2, \ldots, p_N)$ and $\boldsymbol{q} := (q_1, q_2, \ldots, , q_N)$, $j = 1, \ldots, N$. We use finite elements which satisfy the boundary conditions (3) and (4, last line) at the left end of each line, i. e.

$$\psi_0(0) = \varphi_0(0) = 1, \qquad \boldsymbol{\psi}(0) = \boldsymbol{\varphi}(0) = 0.$$

The weak formulation of the boundary value problem (1) then yields the initial value problem [4]

$$\underbrace{\begin{pmatrix} \boldsymbol{L}\boldsymbol{M}_{\psi,\varphi} & 0 \\ 0 & \boldsymbol{C}\boldsymbol{M}_{\psi,\varphi}^\top \end{pmatrix}}_{=:\,\mathcal{M}} \begin{pmatrix} \dot{\boldsymbol{q}} \\ \dot{\boldsymbol{p}} \end{pmatrix} + \underbrace{\begin{pmatrix} \boldsymbol{R}\boldsymbol{M}_{\psi,\varphi} & -\boldsymbol{K}_\psi \\ -\boldsymbol{K}_\varphi & \boldsymbol{G}\boldsymbol{M}_{\psi,\varphi}^\top \end{pmatrix}}_{=:\,\mathcal{K}} \begin{pmatrix} \boldsymbol{q} \\ \boldsymbol{p} \end{pmatrix} + \quad (9c)$$

$$+ \begin{pmatrix} (L\dot{\lambda}_1 + R\lambda_1)\boldsymbol{b}_{\varphi,\psi} \\ (C\dot{u}_1 + Gu_1)\boldsymbol{b}_{\psi,\varphi} \end{pmatrix} - \begin{pmatrix} u_1\boldsymbol{b}_{\psi,\psi'} \\ \lambda_1\boldsymbol{b}_{\varphi,\varphi'} \end{pmatrix} - \begin{pmatrix} u_1\psi(0) - u_2\psi(1) \\ (\lambda_1\varphi(0) + \lambda_2\varphi(1)) \end{pmatrix} = 0$$

with

$$\boldsymbol{M}_{\psi,\varphi} = \int_0^1 \boldsymbol{\psi}\boldsymbol{\varphi}^\top\,dz, \quad \boldsymbol{K}_\psi = \int_0^1 \boldsymbol{\psi}'\boldsymbol{\psi}^\top\,dz, \quad \boldsymbol{K}_\varphi = \int_0^1 \boldsymbol{\varphi}'\boldsymbol{\varphi}^\top\,dz,$$

$$\boldsymbol{b}_{\varphi,\psi} = \int_0^1 \varphi_0\boldsymbol{\psi}\,dz, \quad \boldsymbol{b}_{\psi,\varphi} = \int_0^1 \psi_0\boldsymbol{\varphi}\,dz,$$

$$\boldsymbol{b}_{\psi,\psi'} = \int_0^1 \psi_0\boldsymbol{\psi}'\,dz, \quad \boldsymbol{b}_{\varphi,\varphi'} = \int_0^1 \varphi_0\boldsymbol{\varphi}'\,dz.$$

Together with the first part of the network equations (8)

$$\begin{aligned} \jmath_V + \lambda_1 &= 0, && (9d) \\ \widetilde{C}\dot{u}_2 + \lambda_2 &= 0, && (9e) \\ u_1 - V(t) &= 0, && (9f) \end{aligned}$$

the ADAE system (9) defines a linear system of differential-algebraic equations:

$$\boldsymbol{\mathcal{A}}\dot{\boldsymbol{x}} + \boldsymbol{\mathcal{B}}\boldsymbol{x} = \boldsymbol{f}(t), \qquad \boldsymbol{x} = (\boldsymbol{q}, \boldsymbol{p}, \boldsymbol{u}, \jmath_V, \boldsymbol{\lambda}). \quad (10)$$

The index of (10) can be computed via the generalized eigenvalue problem for the matrix pencil $\mu\boldsymbol{\mathcal{A}} + \boldsymbol{\mathcal{B}}$. We get five ∞-eigenvalues, but have only four algebraic equations (9a,9b,9d,9f). Hence the index is larger than one.

One differentiation of the algebraic equations leads to

$$\widetilde{\boldsymbol{\mathcal{A}}}\dot{\boldsymbol{x}} + \widetilde{\boldsymbol{\mathcal{B}}}\boldsymbol{x} = \widetilde{\boldsymbol{f}}(t) \quad (11)$$

with

$$\widetilde{\boldsymbol{f}}(t) = (0, \ldots, 0, \dot{V}(t))^\top,$$

$$\widetilde{\mathcal{A}} = \begin{pmatrix} LM_{\psi,\varphi} & 0 & 0 & 0 & 0 & Lb_{\varphi,\psi} & 0 \\ 0 & CM_{\psi,\varphi}^{\top} & Cb_{\psi,\varphi} & 0 & 0 & 0 & 0 \\ 0 & -e_n^{\top} & 0 & 1 & 0 & 0 & 0 \\ -e_n^{\top} & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & \widetilde{C} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix},$$

$$\widetilde{\mathcal{B}} = \begin{pmatrix} RM_{\psi,\varphi} & -K_{\psi} & -\psi(0) - b_{\psi,\psi'} & -\psi(1) & 0 & Rb_{\varphi,\psi} & 0 \\ -K_{\varphi} & GM_{\psi,\varphi}^{\top} & Gb_{\psi,\varphi} & 0 & 0 & -(\varphi(0) + b_{\varphi,\varphi'}) & -\varphi(1) \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

This linear-implicit system contains only one algebraic equation. Since the matrix pencil $\mu\widetilde{\mathcal{A}} + \widetilde{\mathcal{B}}$ has only one ∞-eigenvalue, the index of our ADAE system (8) is two, *for all* $N \in \mathbb{N}$. The currents $\lambda_1 = -\jmath_V$ through the voltage source are index-2 variables, as for VC loops *without* considering transmission lines effects. Using linear finite elements, semidiscretization applied to the original formulation (1) of the telegrapher's equations *acts like a deregularization of the PDAE model.*

Conclusion

Using generalized network models for interconnects, the DAE network equations are generalized to a PDAE model. Inspecting the VC loop, we have seen that this ansatz may regularize DAE network equations of higher index. Using semidiscretization w.r.t. space, the method-of-lines approach converts this PDAE model into an approximative DAE system in time only. To reflect the physical properties of the original PDAE network model, the ADAE system should neither be more nor less sensitive, and the regularizing effect should be kept. However, semidiscretization may deregularize a regularized model, if the method-of-lines approach is not consistent with the information flow.

References

[1] Arnold, M.: *A note on the uniform perturbation index.* Rostock. Math. Kolloq. 52 (1998), 33–46.

[2] Campbell, S.L., Marszalek, W.: *ODE/DAE integrators and MOL problems.* Z. f. angew. Math. u. Mech. 76 Suppl. 1 (1996), 251–254.

[3] Campbell, S.L., Marszalek, W.: *The index of an infinite dimensional implicit system.* To appear in Mathematical and Computer Modelling of Dynamical Systems.

[4] Günther, M.: *A joint DAE/PDE model for interconnected electrical networks.* To appear in Mathematical and Computer Modelling of Dynamical Systems.

[5] Günther, M., Feldmann, U.: *CAD based electric circuit modeling in industry. I: Mathematical structure and index of network equations. II: Impact of circuit configurations and parameters.* Surv. Math. Ind. 8 (1999), 97–157.

[6] Günther, M., Rentrop, P.: *PDAE-Netzwerkmodelle in der elektrischen Schaltungssimulation.* Submitted.

[7] Günther, M., Wagner,Y.: *Index Concepts for Linear Mixed Systems of Differential-algebraic and Hyperbolic-Type Equations.* Submitted.

[8] Lucht, W., Strehmel, K., Eichler-Liebenow, C.: *Indexes and special discretization methods for linear partial differential algebraic equations.* BIT 39 (1999), 484–512.

[9] Tischendorf, C.: *Topological index calculation of differential-algebraic equations in circuit simulation.* Surv. Math. Ind. 8, 187–199 (1999)

[10] Wagner, Y.: *A further index concept for PDAEs of hperbolic type.* Submitted to Third IMACS Symposium on Mathematical Modelling in Vienna.

MODELLING NONDETERMINISTIC DISCRETE-EVENT BEHAVIOUR BY DESCRIPTOR SYSTEMS

D. Franke
University of the Federal Armed Forces
Department of Electrical Engineering
D-22039 Hamburg

Abstract. The paper addresses nondeterministic state transitions which are widely encountered in discrete-event systems. Modelling will be based on the arithmetic approach which has been proposed by the author in recent years. It will be shown that the state equations of a nondeterministic finite state machine can always be written as a multilinear descriptor system. Such a model can be used, e.g., to construct the maximum permissive feedback law which enables only deterministic state transitions in a partially nondeterministic plant.

Introduction

Boolean finite state machines are a widely accepted paradigm for modelling discrete-event dynamical systems. The state equations of a deterministic automaton can always be written in an explicit form,

$$\mathbf{x}(k+1) = \mathbf{f}(\mathbf{x}(k), \mathbf{u}(k)), \tag{1}$$

where $\mathbf{x} \in \mathbb{B}^n$ is the Boolean state, $\mathbf{u} \in \mathbb{B}^p$ is the Boolean control, $\mathbf{f}$ is a vector-valued Boolean function, and k is a counter for state transitions [1], [3]. In many applications, however, nondeterministic behaviour is encountered. A finite Boolean automaton is said to be nondeterministic if a given state and input produce a non-unique successor state. Nondeterministic automata have been used in qualitative modelling of continuous dynamical systems [5]. Different from the set oriented approach in [5] the present paper is oriented directly at the state equations which take the *implicit* form

$$\mathbf{f}(\mathbf{x}(k+1), \mathbf{x}(k), \mathbf{u}(k)) = \mathbf{0} \tag{2}$$

in the nondeterministic case. The questions of existence and determination of the solutions of such equations have been studied in [2], on the basis of Boolean algebra.

The problem formulation in modelling is, however, inverse: Let a nondeterministic automaton be given by its transition table or transition graph, then find an equivalent implicit state equation of the type of Eq. (2). It will be shown in the paper that based on arithmetical logic [4], Eq. (2) can allways be written as a multilinear descriptor system,

$$\mathbf{E}(\mathbf{x}(k), \mathbf{u}(k))\,\mathbf{x}(k+1) = \mathbf{g}(\mathbf{x}(k), \mathbf{u}(k)). \tag{3}$$

Arithmetical logic is known from Boolean reliability theory. Any Boolean function $y = f(\mathbf{x})$, $\mathbf{x} \in \mathbb{B}^n$, can be written as a multilinear arithmetic polynomial:

$$\begin{aligned}
n=1: \quad & y = f_0 + f_1 x_1, \\
n=2: \quad & y = f_0 + f_1 x_1 + f_2 x_2 + f_{12} x_1 x_2 = f_0 + f_1 x_1 + x_2 \cdot (f_2 + f_{12} x_1), \\
n=3: \quad & y = f_0 + f_1 x_1 + f_2 x_2 + f_{12} x_1 x_2 + f_3 x_3 + f_{13} x_1 x_3 + f_{23} x_2 x_3 + f_{123} x_1 x_2 x_3 = \\
& \quad = f_0 + f_1 x_1 + f_2 x_2 + f_{12} x_1 x_2 + x_3 \cdot (f_3 + f_{13} x_1 + f_{23} x_2 + f_{123} x_1 x_2),
\end{aligned} \tag{4}$$

etc.

By introducing this notation in the state equation (1), a link can be provided between deterministic automata and classical discrete-time systems theory [4].

Arithmetic descriptor models

The implicit state equations (2) for the nondeterministic case will be considered in the subsequence, in scalar notation:

$$f_i(x(k+1),\ x(k),\ u(k)) = 0, \quad i = 1,...,n. \tag{5}$$

The general multilinear structure according to Eqs. (4) allows to rewrite Eqs. (5) in the following form:

$$\sum_{j=1}^{n} e_{ij}(x(k), u(k))\ x_j(k+1) - g_i(x(k),\ u(k)) = 0, \quad i = 1,...,n \tag{6}$$

where e_{ij} and g_i are multilinear functions. It should be emphasized that multilinearities in terms of the components $x_j(k+1)$, $j = 1,...,n$, do not appear since they would contain only redundant information. Now Eqs. (6) can readily by rewritten in vector notation:

$$\mathbf{E}(x(k), u(k)) \cdot x(k+1) = g(x(k), u(k)). \tag{7}$$

This equation covers the following cases:

(a) Deterministic automaton: $\mathbf{E}(x(k), u(k)) \equiv \mathbf{I}$ (8)

(b) Partially nondeterministic automaton: $\det \mathbf{E}(x(k), u(k)) = 0$ (9)

for those entries $x(k), u(k)$ which produce nondeterministic $x(k+1)$.

(c) Globally nondeterministic automaton: $\det \mathbf{E}(x(k), u(k)) = 0$ (10)

for all entries $u(k), x(k)$.

In the special case of an autonomous system, Eq. (7) takes the form

$$\mathbf{E}(x(k)) \cdot x(k+1) = g(x(k)). \tag{11}$$

The above cases (a), (b), (c) apply accordingly.

Parameter specification

Let a nondeterministic automaton be given by its transition table or transition graph. Then find an equivalent descriptor model (7) or (11), respectively. The procedure will be described w.r.t. Eq. (11). The extension to Eq. (7) can be made accordingly.

Multilinear square matrix $\mathbf{E}(x)$ and vector $g(x)$ take the form

$$\mathbf{E}(x) = \mathbf{E}_0 + \mathbf{E}_1\ x_1 + \mathbf{E}_2\ x_2 + \mathbf{E}_{12}\ x_1 x_2 + \mathbf{E}_3\ x_3 + ... + \mathbf{E}_{12...n} x_1 x_2 ... x_n, \tag{12}$$

$$g(x) = g_0 + g_1 x_1 + g_2 x_2 + g_{12}\ x_1 x_2 + g_3\ x_3 + ... + g_{12...n}\ x_1 x_2 ... x_n. \tag{13}$$

Let the 2^n possible states be denoted by $x^{(1)}, x^{(2)}, ..., x^{(2^n)}$. Then for each $x^{(\ell)}$ with *unique* successor state $x^{(m)}$ one has

$$\mathbf{E}\left(\mathbf{x}^{(\ell)}\right)=\mathbf{I}\ ,\quad \mathbf{g}\left(\mathbf{x}^{(\ell)}\right)=\mathbf{x}^{(m)}, \tag{14}$$

and for each $\mathbf{x}^{(\ell)}$ with possible successor states $\mathbf{x}^{(m_1)}$, $\mathbf{x}^{(m_2)}$, ..., $\mathbf{x}^{(m_r)}$ one has

$$\mathbf{E}\left(\mathbf{x}^{(\ell)}\right)\cdot\mathbf{x}^{(m_\nu)}=\mathbf{g}\left(\mathbf{x}^{(\ell)}\right),\quad \nu=1,\ldots,r\,. \tag{15}$$

In view of the special type of multilinear Eqs. (12), (13) it can be seen from Table 1 that the **E** and **g** parameters will be determined in a successive way from linear Eqs. (14) and (15), respectively.

Table 1. Determination of parameters

	...	$x_3(k)$	$x_2(k)$	$x_1(k)$	parameters involved
$\mathbf{x}^{(1)}$	...	0	0	0	$\mathbf{E}_0,\mathbf{g}_0$
$\mathbf{x}^{(2)}$	...	0	0	1	additional $\mathbf{E}_1,\mathbf{g}_1$
$\mathbf{x}^{(3)}$	...	0	1	0	additional $\mathbf{E}_2,\mathbf{g}_2$
$\mathbf{x}^{(4)}$	...	0	1	1	additional $\mathbf{E}_{12},\mathbf{g}_{12}$
$\mathbf{x}^{(5)}$	...	1	0	0	additional $\mathbf{E}_3,\mathbf{g}_3$
$\mathbf{x}^{(6)}$	...	1	0	1	additional $\mathbf{E}_{13},\mathbf{g}_{13}$
$\mathbf{x}^{(7)}$	...	1	1	0	additional $\mathbf{E}_{23},\mathbf{g}_{23}$
$\mathbf{x}^{(8)}$	...	1	1	1	additional $\mathbf{E}_{123},\mathbf{g}_{123}$
⋮		⋮	⋮	⋮	⋮

Eq. (14) will always yield a unique solution for the newly involved parameters. In case of Eq. (15), due to

$$\text{Rank}\ \mathbf{E}\left(\mathbf{x}^{(\ell)}\right)<n\,, \tag{16}$$

degrees of freedom will arise for the newly involved parameters. They should be utilized to make as many as possible **E**-parameters equal to zero to obtain simple expressions. On the other hand, the rank according to (16) should not be made smaller than necessary. Otherwise spurious solutions $\mathbf{x}(k+1)$ may arise which cannot always be prevented to occur.

Example of a nonautonomous system

Let a *partially* nondeterministic automaton with $u\in\mathfrak{B}^1$, $\mathbf{x}\in\mathfrak{B}^2$ be given by its transition table, see Table 2.

Table 2. Example

u(k)	$x_2(k)$	$x_1(k)$	$x_2(k+1)$	$x_1(k+1)$
0	0	0	0	1
0	0	1	1	1
0	1	0	1	0
0	1	1	0	1
1	0	0	0	0
			0	1
			1	0
			1	1
1	0	1	0	0
1	1	0	0	1
1	1	1	1	0

Application of the above successive procedure yields matrix E and vector g in Eq. (7) with the following elements:

$$e_{11}(x,u) = e_{22}(x,u) = 1 - u + x_1\,u + x_2\,u - x_1\,x_2\,u\,,\quad e_{12}(x,u) = e_{21}(x,u) \equiv 0\,,$$

$$g_1(x,u) = 1 - x_2 - u - x_1 x_2 + 2x_2 u - 2x_1 x_2 u\,,\quad g_2(x,u) = x_1 + x_2 - 2x_1 x_2 - x_1 u - x_2\,u + 3\,x_1\,x_2\,u\,.$$

Conclusions

A link has been provided between nondeterministic discrete-event systems and nonlinear discrete-time descriptor systems. Based on arithmetical logic a successive scheme has been presented to determine the parameters of the finite automaton. Applications of such a model for feedback control purposes will be reported in the near future.

References

1. Bochmann, D., Einführung in die strukturelle Automatentheorie. VEB Verlag Technik, Berlin, 1975.
2. Bochmann, D. and Posthoff, C., Binäre dynamische Systeme. R. Oldenbourg, München, Wien, 1981.
3. Booth, T. L., Sequential Machines and Automata Theory. J. Wiley, New York, 1967.
4. Franke, D., Sequentielle Systeme. Vieweg, Braunschweig, 1994.
5. Lunze, J., Künstliche Intelligenz für Ingenieure: Bd. 2. R. Oldenbourg, München, Wien, 1995.

VERIFICATION OF VARIOUS PIPELINE MODELS

Drago Matko[1], Gerhard Geiger[2] and Withold Gregoritza[2]
[1]Faculty of Electrical Eng. University of Ljubljana, Tržaška 25, 1000 Ljubljana Slovenia
[2]Fachhochschule Gelsenkirchen, Neidenburger Str. 10, 45877 Gelsenkirchen, Germany

Abstract. The paper deals with the verification of three pipeline models: the nonlinear distributed parameters model, the linear distributed parameter model and the linear lumped parameters model. All the models were comparatively verified on the basis of the measurements on a real pipeline.

Introduction

The paper deals with three different pipeline models. First, a nonlinear distributed parameters model is given, which is then linearised and its transfer function is presented. The pipeline is represented as a two - port system. Two representations - the hybrid ones which are used in practice - are given. They involve three different transcendent transfer functions which are then approximated by rational transfer functions using a Padé approximation. The derived models describe the pipeline as a lumped parameter system. Due to the approximation of the high frequency gain, the derived models are only valid for a class of models - namely - well damped pipelines. The derived models were comparatively verified on the basis of the measurements on a real pipeline.

The paper is organised as follows: The mentioned pipeline models are presented in the next section. Then the verification of the models is given on the basis of real plant measurements.

Pipeline Models

Non-linear pipeline model with distributed parameters. The non-linear pipeline model with distributed parameters is obtained by using the equations for continuity, momentum and energy. These equations correspond to the physical principles of mass conservation, Newton's second law and energy conservation. Applying these equations leads under the assumptions that the fluid is compressible, viscous, isentropic, homogenous and one-dimensional to the following coupled non-linear set of partial differential equations [1, 2]:

$$\frac{A}{a^2}\frac{\partial p}{\partial t} = -\frac{\partial q}{\partial x} \tag{1}$$

$$\frac{1}{A}\frac{\partial q}{\partial t} + \bar{\rho} g \sin\alpha + \frac{\lambda(q)}{2DA^2\bar{\rho}}q^2 = -\frac{\partial p}{\partial x} \tag{2}$$

where p is the pressure, q is the flow, A the cross-section of the pipeline, a the velocity of sound, $\bar{\rho}$ the constant density of the homogenous fluid, α the pipeline inclination, λ the dimensionless friction coefficient and D the diameter of the pipeline.

The continuity and momentum equations 1 and 2 form a pair of quasilinear hyperbolic partial differential equations in term of two dependent variables, mass flow rate $q(x,t)$ and pressure $p(x,t)$, and two independent variables, distance along the pipeline x and time t. A general solution is not available; however, a transformation into four ordinary differential equations grouped to two pairs of equations by the characteristics method is possible [3].

Linear pipeline model with distributed parameters. Nonlinear Eqns.(1, 2) are linearised and written in a form using notations common in the analysis of electrical transmission lines. Also, the gravity effect can be included into the working point so $\alpha = 0$ is supposed. The corresponding system of linear

partial differential equations is

$$L\frac{\partial q}{\partial t} + Rq = -\frac{\partial p}{\partial x} \quad (3)$$

$$C\frac{\partial p}{\partial t} = -\frac{\partial q}{\partial x} \quad (4)$$

where $L = \frac{1}{A}$, $R = \frac{\lambda(\bar{q})\bar{q}}{A^2\rho D}$ ($\bar{q}$ is the flow at the working point) and $C = \frac{A}{a^2}$ are the inertance (inductivity), resistance and capacitance per unit length, respectively. Introducing the *characteristic impedance* $Z_K = \sqrt{\frac{Ls+R}{Cs}}$ and $n = \sqrt{(Ls+R)\cdot Cs}$ the linearised model of the pipeline can be written in one of the following two causal forms which differ from each other with respect to the model inputs (independent quantities) and outputs (dependent quantities) and where P and Q are the double Laplace transformations of the pressure p and flow q respectively and indexes $_0$ and $_L$ denote the inlet and the outlet of the pipeline respectively.

1. **Inputs Q_0, P_L, outputs Q_L, P_0:**

$$Q_L = \frac{1}{\cosh(nL_p)}Q_0 - \frac{1}{Z_K}\tanh(nL_p)\cdot P_L \quad (5)$$

$$P_0 = Z_K\tanh(nL_p)Q_0 + \frac{1}{\cosh(nL_p)}\cdot P_L \quad (6)$$

2. **Inputs Q_L, P_0, outputs Q_0, P_L:**

$$Q_0 = \frac{1}{\cosh(nL_p)}Q_L + \frac{1}{Z_K}\tanh(nL_p)\cdot P_0 \quad (7)$$

$$P_L = -Z_K\tanh(nL_p)Q_L + \frac{1}{\cosh(nL_p)}\cdot P_0 \quad (8)$$

Linear pipeline model with lumped parameters. The pipeline as a lumped parameter system can be presented as a second order transfer function in the form

$$G(s) = \frac{b_2s^2 + b_1s + b_0}{a_2s^2 + a_1s + 1}e^{-sT_d} \quad (9)$$

where T_d is the dead time. The transcendent transfer functions are approximated by a rational transfer function with dead time however only for a class of well damped pipelines. The parameter b_0, i.e. the static gain of the transfer function is obtained from the first term of the Taylor Series expansion of the transcendent function. The term $a_2 = \frac{1}{\omega_0^2}$ is determined from the eigen-frequency ω_0 of the pipeline, which can be interpreted as follows: the shock wave originating at one end of the pipeline returns after reflection at the other end of the pipeline with the opposite phase. The half period of the oscillations is consequently equal to the time needed by the shock wave to travel along the pipeline and back. This gives the radial eigen-frequency

$$\omega_0 = \frac{\pi}{2\sqrt{LC}}\cdot\frac{1}{L_p} \quad (10)$$

Next, the high frequency gain is approximated from transcendent function under the assumption of well damped pipeline ($\sqrt{\frac{C}{L}}\frac{RL_p}{2} >> 1$) and known dead time is applied. In this way, four coefficients of the transfer function (9) are determined. The remaining two are obtained by a Padé approximation of the transcendent transfer function. Since the high frequency gain approximation is valid only under certain conditions, the derived models are only valid for one class of pipelines. The parameter a_2 is the same for all transfer functions, the remaining parameters are as follows

1. **Pressure → pressure and flow → flow transfer function** $G_0(s) = \frac{1}{\cosh(nL_p)}$. Since this transfer function connects quantities at different ends of the pipeline, the dead time for (9) is known - it is the time needed for the shock wave to travel along the pipeline.$T_d = \sqrt{LC}L_p$ The static gain

of G_0 is 1, so the coefficient $b_0 = 1$. The high frequency gain is determined by $\lim_{\omega\to\infty} |G_0(j\omega)| = \lim_{\omega\to\infty} G(s) = \frac{b_2}{a_2} \approx 2e^{-\sqrt{\frac{C}{L}}\cdot\frac{RL_p}{2}}$ yielding $b_2 = \frac{8}{\pi^2}LCL_p^2e^{-\sqrt{\frac{C}{L}}\cdot\frac{RL_p}{2}}$ The remaining coefficients b_1 and a_1 are determined by a Padé approximation of $G_0(s)$:

$$b_1 = \frac{L_pC\left(96L - 192Le^{-\sqrt{\frac{C}{L}}\cdot\frac{RL_p}{2}} - 24\pi^2L - \pi^2L_p^2R^2C + 12\pi^2L_pR\sqrt{LC}\right)}{12\pi^2(L_pRC - 2\sqrt{LC})} \tag{11}$$

$$a_1 = \frac{L_pC\left(96L - 192Le^{-\sqrt{\frac{C}{L}}\cdot\frac{RL_p}{2}} + 5\pi^2L_p^2R^2C - 12\pi^2L_pR\sqrt{LC}\right)}{12\pi^2(L_pRC - 2\sqrt{LC})} \tag{12}$$

2. **Pressure → flow transfer function** $G_0(s) = \frac{1}{Z_k}\tanh(nL_p)$. The dead time of the treated transfer function is zero ($T_d = 0$) since it connects a change of the flow at one end of the pipeline if the pressure changes at the same end. The static gain of $G_0(s)$ is zero, so the coefficient $b_0 = 0$. The high frequency gain is determined by $\frac{b_2}{a_2} = \lim_{\omega\to\infty}\left|\frac{1}{Z_k}\tanh(nL_p)\right| \approx \sqrt{\frac{C}{L}}$ yielding $b_2 = \frac{4}{\pi^2}L_p^2C\sqrt{LC}$ The remaining coefficients b_1 and a_1 are determined by the same procedure as in the previous case:

$$b_1 = L_p\cdot C \qquad a_1 = \frac{1}{3\pi^2}L_p\left(12\sqrt{LC} + \pi^2L_pRC\right) \tag{13}$$

3. **Flow → pressure transfer function** $G_0(s) = Z_k\tanh(nL_p)$. The dead time of the treated transfer function is zero ($T_d = 0$), since it connects a change of the pressure at one end of the pipeline if the flow changes at the same end. The static gain of $G_0(s)$ is L_pR, yielding $b_0 = L_pR$. The high frequency gain is determined by $\frac{b_2}{a_2} = \lim_{\omega\to\infty}\left|Z_k\tanh(nL_p)\right| \approx \sqrt{\frac{L}{C}}$ yielding $b_2 = \frac{4}{\pi^2}L_p^2L\sqrt{LC}$. The remaining coefficients b_1 and a_1 are determined by the same procedure as in the two previous cases:

$$b_1 = \frac{L_p(288L_p^2R^2LC + \pi^2L_p^4R^4C^2 - 288L_pRL\sqrt{LC} - 72\pi^2L^2)}{24\pi^2(L_p^2R^2C - 3L)} \tag{14}$$

and

$$a_1 = \frac{L_p(96L_pRLC - 96L\sqrt{LC} - 16\pi^2L_pRLC + 3\pi^2L_p^3R^3C^2)}{8\pi^2(L_p^2R^2C - 3L)} \tag{15}$$

Experimental verification of pipeline models

The nonlinear and both linear (distributed and lumped parameter) models were verified on the basis of measurements on a real pipeline. The relative roughness, which is the most uncertain parameter of the model was estimated on the basis of the static drop of the pressure in the pipeline.

Fluid transients were generated for experimental verification by closing and opening of a valve at the beginning of the pipeline. This leads to approximately 35 % excitation of flow transients and 4 % excitation of the pressure transients. There were no controllers for flow rate or pump pressure.

Pressures and flows on both sides of the pipeline were measured. Pressure at the input and flow at the output of the pipeline were chosen as model inputs. Flow at the input and pressure at the output of the pipeline were calculated with the program PIPESIM (nonlinear model), with the convolution of the impulse responses of the transcendent transfer functions and model inputs according to Eqns. (7) and (8) and with the simulation of the rational transfer functions.

Fig. 1 depicts the measured pressure (left) and flow (right) and corresponding responses of the three treated models (nonlinear, linear with distributed parameters and linear with lumped parameters).

If plotted in the small scale no differences among the four responses could be noticed, so in Figure 2 a detail of the pressure response is shown in large scale. It can be seen that the responses of the nonlinear and linear distributed parameter models coincide very well with measurements, while the responses of the linear lumped parameter model exhibit some deviations. This is due to the violation of the supposition of well damped pipeline.

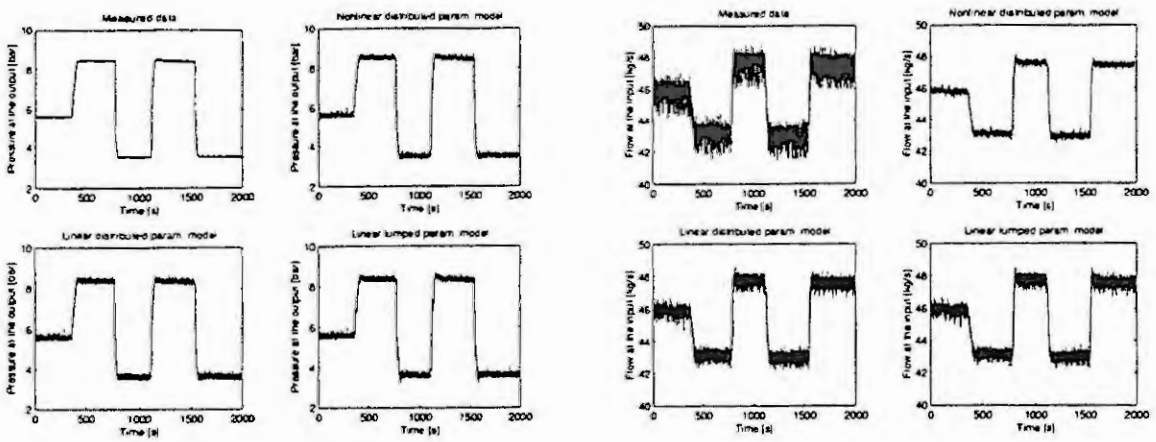


Figure 1: Verification of the models - the pressure responses (left) and the flow responses (right)

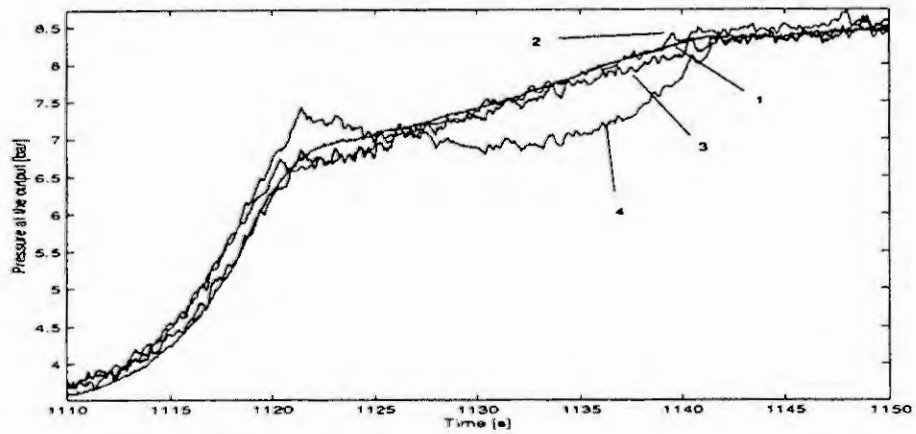


Figure 2: Verification of the models - the detail of the pressure responses (1: measured data, 2: nonlinear distributed parameters model, 3: linear distributed parameters model, 4: linear lumped parameters model)

Conclusion

Three models of the pipeline: the nonlinear distributed parameters model, the linear distributed parameters model and the linear lumped parameters model were verified on the basis of the measurements on a real pipeline. The nonlinear distributed parameters model was simulated by a special program PIPESIM and serves the best results, however has the highest computational demands. With small changes of the signals around a working point the linear distributed parameter model, having a transcendent transfer function, serves with less computational demand nearly as good results as the nonlinear model. The linear lumped parameters model, having a rational transfer function and thus lowest computational demand, can be derived with a certain supposition, which was not fulfilled in the treated case, but in spite of this supposition violation the results of the simulation are acceptable for some purposes such as e.g. controller design.

References

1 J.D.Anderson. Basic Philosophy of CFD. In J.F.Wendt, editor, *Computational Fluid Dynamics*. Springer-Verlag, Berlin Heidelberg, 2nd edition, 1996.

2 V.L.Streeter, E.B.Wylie. *Fluid Mechanics*. McGraw-Hill, New York, 8th edition, 1985.

3 V.L.Streeter, E.B.Wylie. *Fluid Transients in Systems*. Prentice-Hall, London, 1993.

THE VALIDATION OF COMPUTER MODELS OF A MECHANICAL VENTILATOR AND THE HUMAN RESPIRATORY SYSTEM INTENDED FOR USE IN ADULT INTENSIVE CARE

C. M. Murphy[1], B. Brook[2], D. G. Tilley[1] A. W. Miles[1], D. Breen[3] and A. Wilson[2]
[1]Department of Mechanical Engineering, University of Bath.
[2]Department of Medical Physics and Clinical Engineering, University of Sheffield.
[3]Royal Hallamshire Hospital, Sheffield.

Abstract. The case for developing a mathematical model of a mechanical ventilator is established in the context of building a clinical decision support tool for use by those prescribing ventilator therapy on hospital Intensive Therapy Units. The basis of such a model is described and implemented in the Bath*fp* simulation package. The algorithm implemented by the ventilator to control the flow rate of gas to and from the patient is not known *a priori*. However a system for approximating and calibrating this algorithm is presented and applied to produce a model of an Evita II ventilator which mimics the operation of the ventilator under controlled conditions.

Introduction

Critical care is an area of medicine which makes extensive use of artificial ventilators. Clinicians on Intensive Therapy Units (ITUs) prescribe their use to treat a variety of life-threatening conditions. To secure a successful recovery, the clinician typically aims to achieve certain physiological goals within a given period of time. It is proposed that a clinical decision support tool may be built to back up the expert judgement of the clinician, by allowing the efficacy of a proposed therapy to be tested on a computer simulation before being prescribed for the patient. The feasibility of building a computer model of the human respiratory system which is suitable for estimation of its response to artificial ventilation is being investigated. To this end, data is being recorded from suitable patients on the ITU at the Royal Hallamshire Hospital in Sheffield who are being invasively monitored as a routine part of their care. These data will be used to calibrate and validate mathematical models of the critical aspects of the human respiratory system. For such a model to accurately predict the reaction of the respiratory system to various artificial ventilation regimes, a model of the ventilator which can couple with the respiratory model is also required.

Mathematical models are being implemented in the Bath*fp* dynamic simulation package developed at the University of Bath. This package has previously been applied to respiratory system simulation [2], where it was used to model diving equipment and their interaction with the human physiology. However this existing respiratory model is not suitable for simulation of patients on an ITU since it was designed to emulate typical divers in excellent physical condition, rather than those with critical illness. Additionally the model depended on certain physiological parameters which are difficult or impossible to measure or calculate in the operational clinical environment. Therefore a simplified and rigorously-validated respiratory model is ultimately needed if it is to be used to simulate the ventilator and respiratory system on the ITU.

The ventilator

The ventilator used for the work here is the Dräger Evita II. This is responsible for delivering gas to the patient and controlling the pressure applied to the airway. Gas is introduced into the circuit at the *mixer* valve which is individually calibrated to deliver a required flow rate. The ventilator can vent gas from the circuit through the *PEEP/PIP* (positive end expiratory pressure/peak inspiratory pressure) valve, and thus limit airway pressure. Gas flows into or out of the lungs of the patient via an endo-tracheal (ET) tube. This tube is connected to a component called the y-piece which allows gas to flow between the ET tube, the mixer valve and the PEEP/PIP valve. The y-piece is connected to the ventilator valves with convoluted plastic hoses. A humidifier unit is connected in series between the mixer valve and the y-piece. The standard ventilator circuit as represented by Bath*fp* is shown in Figure 1 attached to a model of the human respiratory system.

The Evita II ventilator has several modes of operation. We are concerned only with the Biphasic Positive Airway Pressure (BiPAP) mode initially. The clinician has control over the pressures applied during inspiration and expiration, the frequency of breaths, the ratio between inspiratory and expiratory times along with the time taken for the pressure to ramp between the expiratory and inspiratory pressures at the start of an inspiration. These determine what the *demand* pressure is at any given time during a breath, which can be compared to the

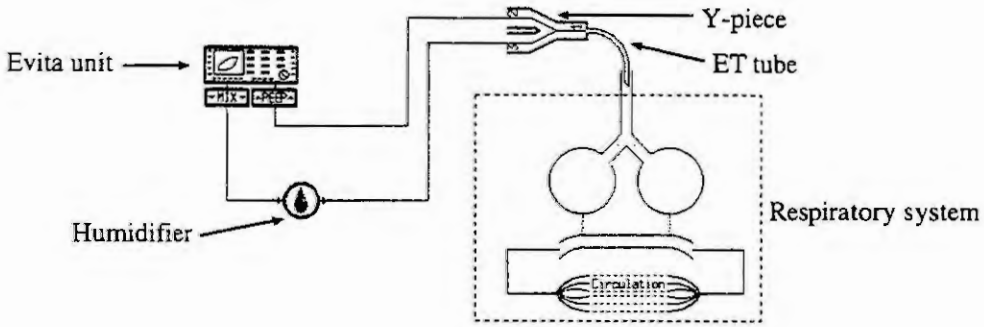


Figure 1: A ventilator and respiratory model circuit.

actual airway pressure. The flow rate at the mixer valve or the status of the PEEP/PIP valve can then be changed accordingly. Note that in BiPAP mode there is no way to explicitly control flow rate into or out of the patient, or the volume of gas received by the patient; flow results from the difference between the prescribed airway pressure and the intra-lung pressure.

Modelling of the pneumatic components of the ventilator is achieved by assuming uniform gas characteristics within each of a set of discrete vessels. These are connected together by ports so that there is a path from each vessel into at least one other vessel where flow between them is controlled via a model of a pneumatic orifice (for example see [1]). A friction model is used to account for the convoluted nature of the hoses (see [3] and [4]). The simulation integrates the rate of change of the pressure (P in Pa) and temperature (T in K) of the gas in each component along with the mass of each constituent gas (m_i in kg). The method for this model is adapted from [2]. The method depends on the ideal gas law $P = \rho RT = \frac{mRT}{V}$ where ρ, m and R are the density (kg/m^3), mass (kg) and gas constant (J/kg.K) of the gas and V is the volume of the vessel in m^3. The rates of change of the parameters of the gas in each vessel are given in terms of $q_{in,j}$, the flow rate into the vessel at port j in kg/s; $T_{in,j}$, the temperature of the gas flowing into the vessel at port j in K; m_i^f, the proportion by mass of the ith component gas in the vessel; and $m_{in,j,i}^f$, the proportion by mass of the ith component of the gas coming into the vessel at the jth port. The derivatives are as follows

$$\dot{P} = \frac{\dot{m}RT + m\dot{R}T + mR\dot{T}}{V};$$

$$\dot{T} = \frac{1}{m}\sum_{j=1}^{N_{ports}} \phi_j \text{ where } \phi_j = \begin{cases} q_{in,j}(T_{in,j} - T) & \text{if } q_{in,j} > 0 \\ 0 & \text{if } q_{in,j} \le 0 \end{cases}; \quad \dot{m}_i = \sum_{j=1}^{N_{ports}} \psi_{i,j} \text{ where } \psi_{i,j} = \begin{cases} q_{in,j} m_{in,j,i}^f & \text{if } q_{in,j} > 0 \\ -q_{in,j} m_i^f & \text{if } q_{in,j} \le 0. \end{cases}$$

The gas constant R is given as $R = \frac{R_0}{M}$ where R_0 is the universal gas constant, equal to 8.31451 J/K.kg and M is the molecular mass of the gas in kg/mol which is defined thus

$$M = m\sum_{i=1}^{N_{gases}} \frac{M_i}{m_i}; \quad \text{and hence} \quad \dot{R} = \frac{R_0}{m^2}\sum_{i=1}^{N_{gases}} \frac{\dot{m}_i m - m_i \dot{m}}{M_i}$$

where M_i is the molecular mass, in kg/mol, of the ith gas in the mixture.

Ventilator algorithm modelling and calibration

The ventilator has control over the flow rate delivered to the breathing circuit at the mixer valve along with the state of the PEEP/PIP valve. Certain information regarding the operation of the ventilator was supplied by the manufacturers. However this information was incomplete and certain assumptions and deductions had to be made in order for a model to be built. The rate of change, or slew, of the flow rate delivered to the ventilator circuit at the mixer valve ($\dot{Q}$) can be changed by up to 8 L/m per 6 ms machine timestep, to correct discrepancies between the demand pressure (as calculated from the pressure and frequency settings made by the clinician) and the actual airway pressure. During the inspiration phase, if the airway pressure exceeds the demand pressure by 25 mbar or more, the PEEP/PIP valve opens to vent the pressure from the circuit. During expiration, the mixer valve shuts off and the PEEP/PIP valve opens until the expiratory pressure is reached.

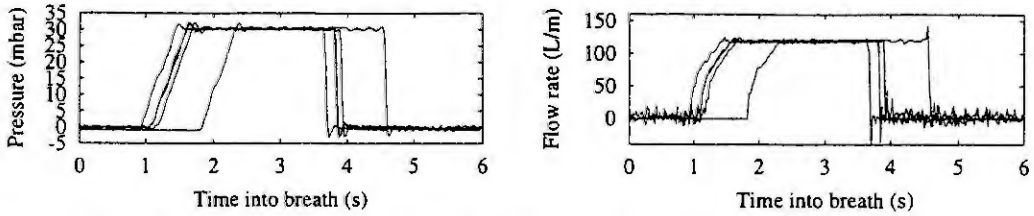


Figure 2: Waveforms for circuit pressure and flow rate out of the circuit measured with a fixed resistance load.

The Evita must implement some algorithm for deciding on $\dot{Q}$, the slew of the flow rate at the mixer valve, to account for airway pressure discrepancies. It is not strictly necessary to reproduce the algorithm exactly, as long as the waveforms of pressure and flow rate experienced by the patient mimic those achieved experimentally. A sensible basis for the calculation algorithm of $\dot{Q}$ is $\Delta P = P_d - P_a$, where P_d is the demand pressure, based on the front-panel settings of the ventilator, and P_a is the airway pressure as measured by the ventilator. We know $\dot{Q}_{max}$, the maximum that the ventilator can slew the flow rate by in any timestep. Here, a simple three-part linear relationship is adopted, so that

$$\dot{Q} = \begin{cases} \dot{Q}_{max} & \text{if } \dot{Q}_{max} \leq (\Omega \times \Delta P) \\ \Omega \times \Delta P & \text{if } -\dot{Q}_{max} < (\Omega \times \Delta P) < \dot{Q}_{max} \\ -\dot{Q}_{max} & \text{if } (\Omega \times \Delta P) \leq -\dot{Q}_{max} \end{cases}$$

where Ω is a parameter (in L/m per timestep) whose value is sought.

It is important to be able to calibrate and validate the ventilator in isolation from the model of the respiratory system, so that the calibration of the ventilator is independent to the calibration of the elastic lungs. Simple loads may be placed on the breathing circuit in place of the human being. These can be simple pneumatic resistors or pressure-related flow cut-off valves which allow gas to escape to the atmosphere, rather than flow into or out of the patient. In a controlled experiment, some such resistor is placed at the port of the y-piece which normally connects to the patient to allow gas to escape the circuit to atmosphere. A pneumotachograph is inserted between the y-piece and the resistor to measure the flow rate of gas to the atmosphere. Additionally a pressure transducer is inserted into the circuit near the mixer valve. For each of several resistors, the ventilator may be put into various modes of operation by varying the inspiratory pressure and ramp time and recording the measured pressures and flow rates for several experimental *breaths*. The characteristics of the resistor, i.e. the relationship between flow rate and pressure drop, are not known *a priori* but can be deduced if it is assumed that the pressure measured near the mixer valve is closely related to the pressure in the ventilator circuit next to the resistor. Taking all experiments involving a given resistor, the flow rate may be regressed against pressure drop to derive a model which approximates that resistor.

Using a variety of resistors will allow the model parameters, and specifically Ω, to be varied to find the value which results in a model which most accurately simulates the experimental data. This has been carried out using a ventilator connected to four such resistive components in turn, varying the inspiratory pressure between 10, 20 and 30 mbar and the ramp time between 0.5 and 1 s. For each resistor, all combinations which did not result in the ventilator entering its alarm mode were used, with the expiratory pressure held at 0 mbar and the breath frequency set to 10 breaths per minute. The waveforms for a typical configuration, with the inspiratory pressure of 30 mbar and a ramp up time of 0.5 s is shown in Figure 2.

A model of the experiment was built in Bath*fp* and simulations carried out. The effect of varying Ω is of most interest, as accurate simulation of the ventilator algorithm is crucial to the success of the proposed model of the human respiratory system when attached to the ventilator. Figure 3 shows the simulation of the configuration shown in Figure 2 over a series of three different values of Ω. These illustrate that if Ω is too low ($\Omega = 100$) then the ventilator is sluggish in producing the demand pressure while if it is too high ($\Omega = 6000$), overcorrection occurs and oscillations result. A value of Ω in the region of 2000 most faithfully reproduces experimental waveforms.

In addition to validating the value for Ω against the other resistors and other values of pressure and ramp time, this concept of independent calibration can be extended to data collected during ventilation of real patients. The uncertainty introduced by the need to model the dynamics of the lung can be mitigated by using a similar principle to the one used to decide the characteristics of the resistors in the first phase of the calibration. Here the relationship between the pressure in the ventilator circuit and the flow rate into the trachea is more complicated

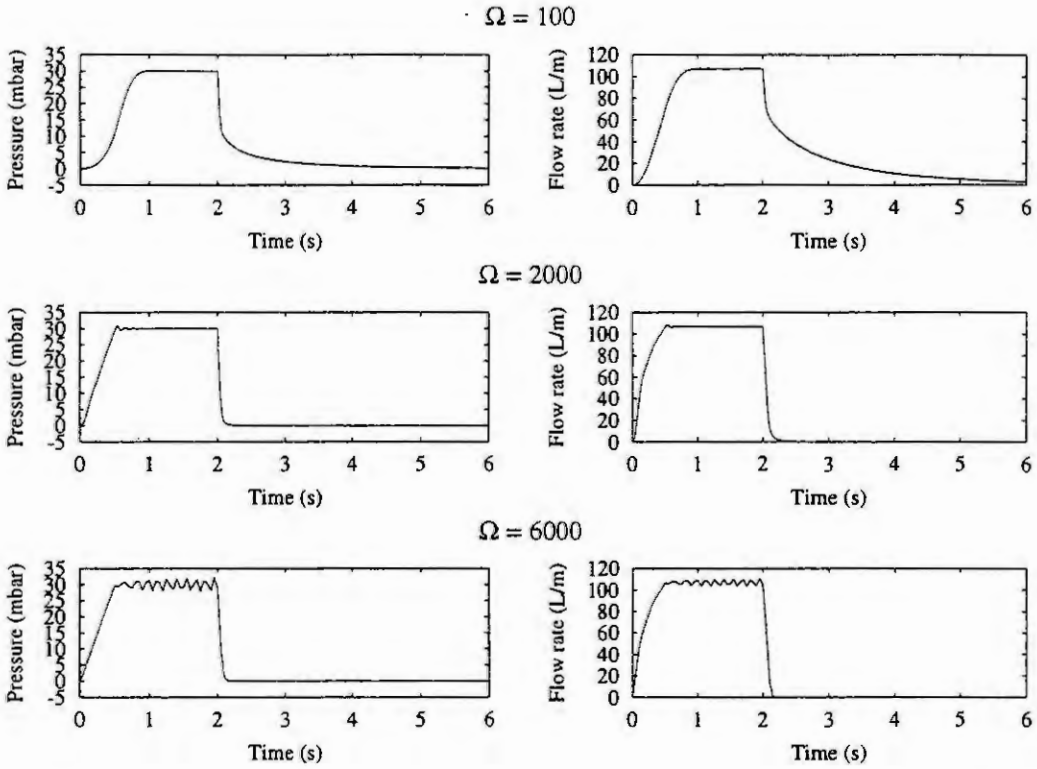


Figure 3: Simulated circuit pressure and flow rate through the resistor with three values of Ω.

due to the hysteresis but preliminary investigation has determined that the relationship is sufficiently well-defined for a simple model of flow rate in terms of circuit pressure to be produced. Hence the requirement for an explicit lung elastance model is abolished.

Conclusions

A mathematical model of a mechanical ventilator has been calibrated under controlled conditions without sophisticated modelling of a load. Once calibrated using the above methods, it is planned that the resulting model of the ventilator will be sufficiently validated for employment in conjunction with a suitable model of the human respiratory system for use in a clinical decision support tool. Work is continuing to develop such a model.

We wish to thank Dave Gaulton of Dräger for supplying information on the operation of the Evita ventilator. The work described in this paper was made possible by the support of the Engineering and Physical Sciences Research Council, grant numbers GR/L74323 and GR/L74835.

References

[1] Blaine W Andersen. *The analysis and design of pneumatic systems*. Robert E. Kreiger Publishing Company, Malabar, Florida, 1985.

[2] J K W Lo. *Mathematical modelling of mixed gas breathing equipment and associated systems*. PhD thesis, University of Bath, 1995.

[3] B S Massey. *Mechanics of fluids*. Van Nostrand Reinhold, 6th edition, 1989.

[4] D G Tilley, S P Tomlinson, and J Livesey. Computer simulation of a semi-closed-circuit breathing system. *Proc Instn Mech Engrs*, 205:163–173, 1991.

Use of the MATLAB Non-Linear Identification Tool to Optimize Parameter Estimates in a Dynamic Response of Two-Stage Pressure Relief Valve Model

Dr.S.P.Tomlinson
Underwater Systems Integration, Defence Evaluation & Research Agency
Winfrith Technology Centre,
Dorchester DT2 8XJ, United Kingdom

Dr.A.Bozin
Cambridge Control Ltd (A MathWorks Company)
Matrix House, Cowley Park,
Cambridge CB4 0HH, United Kingdom

Abstract

This paper summarizes the analysis undertaken at the Defence and Evaluation Research Agency (DERA Winfrith) and Cambridge Control Ltd to optimize parameter estimation of models of the dynamic response of fast acting hydraulic pressure relief valves. This valve forms a safety feature in the hydraulic transmission for a Towed Acoustic Generator (TAG) used by the British Navy for mine detonation purposes. The relatively high and complex signal frequency range requires a highly responsive relief valve to cope with close proximity mine detonations capable of transmitting 35G loadings. Conventional relief valves are inadequate and an extensive design and test programme was devised to optimize the design of a two-stage relief valve capable of meeting the dynamic and steady-state pressure override (increase in inlet pressure with flow rate) requirements of the system. The design study necessitated computer modeling of various fast response valves and circuit configurations. Parameter estimation was a vital part of the simulation process and this was performed using the MATLAB Non-linear Parameter Identification Tool. By using input signals to obtain output response behaviour, it was possible to identify the valve parameters to a high level of accuracy. The parameter estimation is regarded as an important aspect of the validation process when using the system model for prediction purposes.

1 Introduction

Reliable simulation of industrial systems requires not only the correct model but also accurate model parameters. In particular it is recognized that manufactures' component data cannot always be relied upon for system simulation. Cambridge Control has developed a MATLAB-based software tool to estimate component parameters from test data. System identification, also known as model matching, is the use of mathematical techniques to obtain a model and model parameters which give an accurate representation of observed system behaviour. Moreover, identification is a process that involves not only the estimation of parameter values from real plant data, but also collection and data analysis. The software tool developed includes not only a routine for parameter estimation but also data collection and analysis tools to provide an integrated test and identification package. The identification process is much broader than estimating parameters for a given set of data, and affects the modeling, experimentation and validation of systems.

In this paper we present the results of applying the Non-Linear Identification Tool (N[]ID) to identify the unknown (physical) parameters associated with a two-stage relief valve which is used by the British Navy. The remaining part of the paper is organized as follows. A mathematical model of the two-stage

relief valve is given in Section 2. Section 3 gives a brief description of the NLID package. Identification results are presented in Section 4. Finally, some concluding remarks are given in Section 5.

2 Action of a two-stage relief valve

A two-stage relief valve shown in Fig. 1 is a hydraulic device that gives a very precise control of system (inlet) pressure. It maintains this pressure at approximately a set level, ensuring only a small change if the system flow rate increases significantly. The main piston, sensing system pressure on its underside, is held in place by a light spring and is 'pressure balanced' to block the relief path if this pressure is below a desired set level. The system pressure is also sensed by the pilot piston, which acts against a stiff spring. This is pre-loaded to the set level, termed the "cracking pressure". In parallel with the main piston is an orifice (some designs have this orifice drilled through the piston). If the cracking pressure is exceeded, a flow path exists, permitting a small flow to pass through the orifice and out of the pilot stage. This creates a pressure differential across the main piston, unbalancing it and causing it to open. As the main piston acts against a very light spring, the relatively large opening of this piston controls the system pressure to the set cracking level very precisely.

The valve does not respond instantaneously to changes in system pressure due to various dynamic actions in the valve. These are due to the motion of the spool and the build up of pressure at the valve inlet and in the pilot chamber. The valve design was optimized to minimize dynamic reactions in order to ensure very rapid response and accurate steady-state behaviour. The differential and algebraic equations describing system behaviour are given below. The flow rate Q_{pi} from inlet to the pilot chamber at pressure p_2 is given by:

$$Q_{pi} = k_c C_d \pi \frac{D_{sp}^2}{4} \sqrt{\frac{2(p_1 - p_2)}{\rho}}$$

The pilot spring (rate k_{sp}) is assumed to open at the valve cracking pressure p_c and respond instantaneously to changes in pilot chamber pressure p_2. The pilot valve opening y is given by:

$$y = \frac{A_p}{k_{sp}}(p_2 - p_c)$$

where A_p is the pilot poppet frontal area and p_c the cracking preload. Assuming an annular flow area, proportional to the opening y, the flow rate Q_{po} from the pilot chamber p_2 is given by:

$$Q_{po} = k_c C_d \pi D_p y \sqrt{\frac{2(p_2 - p_3)}{\rho}}$$

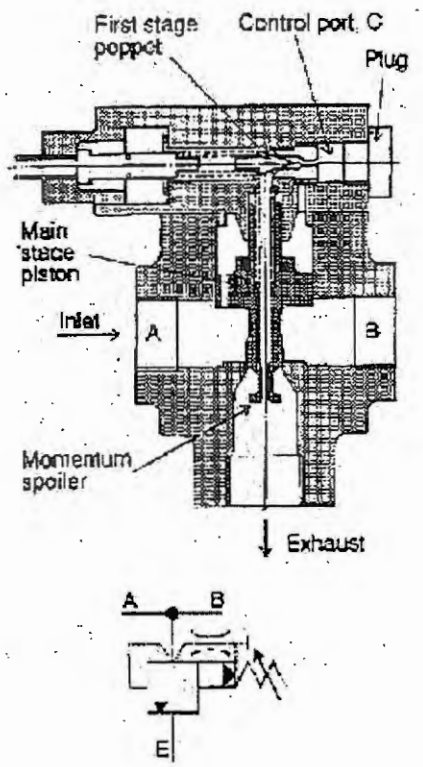


FIGURE 1: Two-stage pressure relief valve.

A flow force F_f acting on the main spool is due to the change in fluid momentum as it passes through the valve (flow area A_f). This is an imprecise term and varies considerably with valve construction. The user is therefore allowed to scale (factor k_f between 0 and 1) the nominal force in accordance with any experimental or design data available.

$$F_f = 0.5\rho v_f Q_1 = \frac{0.5\rho Q_1^2}{k_k C_d A_f}$$

The acceleration of the valve spool is given by:

$$\frac{d^2x_v}{dt^2} = \frac{1}{m}\left(A_{sp}(p_1 - p_2) + k_f F_f - f\frac{dx_v}{dt} - k_s x_v\right)$$

where m is the spool mass, f the viscous friction coefficient and k_s the spring rate. The velocity dx_v/dt and displacement x_v of the spool are obtained by successive integration of this equation. A flow rate term $A_{sp}dx_v/dt$ is created by the spool motion and this has a transient effect on the pilot chamber pressure p_2 acting above the main spool. The chamber pressure p_2 is given by:

$$\frac{dp_2}{dt} = \frac{\beta}{V_2}(Q_{pi} - Q_{po} + A_{sp}\frac{dx_v}{dt})$$

where β is the fluid bulk modulus and V_2 the control volume. The flow area A_f and hence flow rate are thus obtained. Assuming an annular flow area, proportional to the opening x_v, the valve flow rate Q_2 is obtained using the standard orifice equation:

$$Q_2 = \pi D_{sp} x_v \sqrt{\frac{2(p_1 - p_3)}{\rho}}$$

For an inlet flow rate Q_1, the inlet pressure p_1 is given by:

$$\frac{dp_1}{dt} = \frac{\beta}{V_1}(Q_1 - Q_2 - Q_{pi} - A_{sp}\frac{dx_v}{dt})$$

Where V_1 is the inlet control volume.

3 The Matlab non-linear identification tool

NLID package has been developed to perform parameter estimation of non-linear dynamic systems. It provides a Graphical User Interface (see Fig. 2) to assist in physical parameter identification and requires the SIMULINK in order to obtain a dynamic model of the system.

The package allows the user to choose the sub-system for which parameters are to be identified, to select the type of model to be used and to select which parameters are to be identified. The user has to provide initial guesses for the unknown parameters as well as the lower and upper limits; both of these requiring judgment based on user experience (i.e., a priori knowledge). By subtracting simulated from measured output, system identification is formulated as an optimization problem whereby a "merit" or "cost" function, based on this error, is minimized (subject to the parameter bounds).

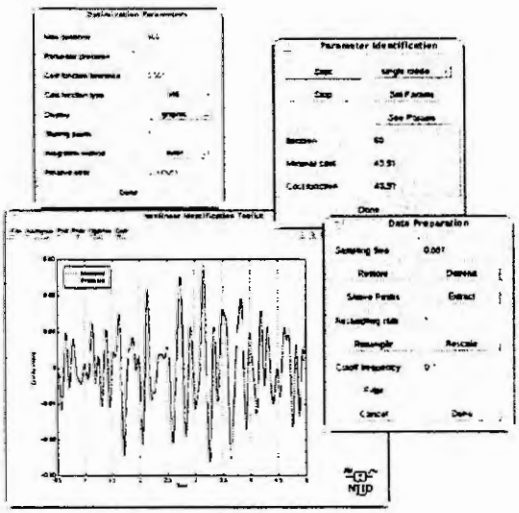


FIGURE 2: Non-linear identification tool.

The output from a parameter identification session is either a MATLAB file or a report containing identified system parameters. In addition, the NLID package allows measured and simulated data to be plotted in order to make a decision whether or not the identified model accurately represents the observed behaviour of the power system.

The user can view the progress of an identification while the identification is running and the final results are available in the MATLAB workspace when an identification is complete. Intermediate results can be plotted after each simulation. The user can terminate the identification before it has completed to retrieve the intermediate results or to change the initial parameter values or the optimization parameters.

4 Parameter estimation results

The two-stage relief valve shown in Fig. 1 is dependent for both its dynamic and steady-state behaviour on a number of design parameters.

Intuitively, rapid dynamic response will be obtained by minimizing the following:

P1 Friction (viscous and Coulomb)

P2 Moving spool/poppet masses

P3 Pilot stage control chamber volume

P4 Inlet control volume

P5 Main stage spool overlap

Low pressure override is also an important feature of valve design and is dependent on:

P6 Pilot and main stage spring rate settings

P7 Pilot chamber inlet and outlet flow areas and discharge coefficients

P8 Main stage flow area and discharge coefficient

P9 Flow force

Some of these parameters are known to a high degree of accuracy whereas others are known less accurately. The latter includes parameters P1, P3, P7 and P9 above. In order to optimize the practical design of the relief valve, it was necessary to measure its dynamic and steady-state response. The response data from the tests was used in the model for parameter identification by N⌊⌋D package.

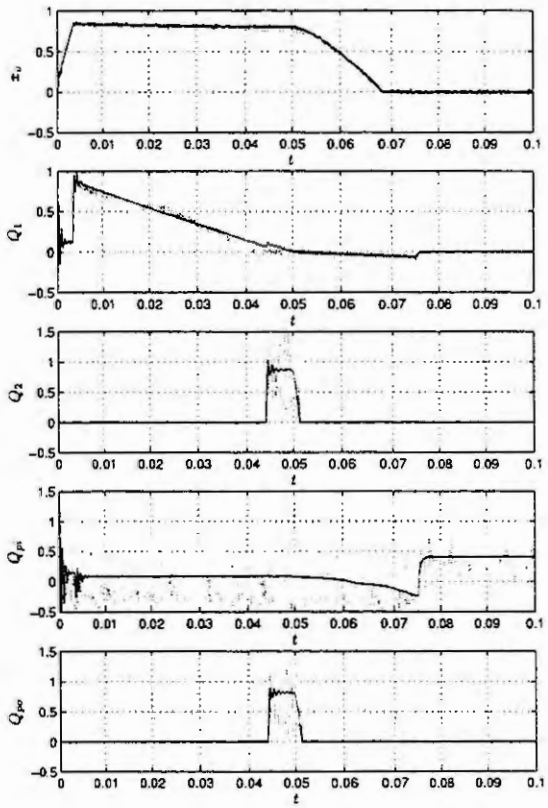


FIGURE 3: Time evolutions of the measured (dashed gray line) and predicted (solid line) outputs.

Figure 3 shows the time evolutions of the measured and predicted (obtained from a two-stage pressure relief valve model given in Section 2) outputs for one particular test. These plots clearly illustrate that the difference between the identified model outputs and the measured outputs are negligible indicating a very good fit to the given data set. It should be mentioned that before the parameter estimation has been attempted, the following scalings have been performed in order to improve the performance of the optimization routine and therefore to increase the accuracy of the estimated parameters:

(1) The parameters are scaled so that they are all of similar magnitude

(2) The measured output signals are normalized to make them equally important

5 Concluding remarks

In this paper we have presented the results of applying the N⌊⌋D tool to get the unknown physical parameters of the two-stage pressure relief valve. In order to achieve that, the mathematical model of the two-stage pressure relief valve has been developed first. A number of tests have been performed on a rig provided by DERA Winfrith to collect data to be used for parameter identification. Preliminary results of applying the N⌊⌋D tool show that it is feseable to get very accurate and realistic parameter estimates associated with the two-stage pressure relief valve model which can be then use to evaluate its performance.

REDUCING THE PARAMETER SPACE OF A NONLINEAR BIOLOGICAL MODEL BY TESTING THE MODEL PURPOSIVENES

M. Zec, N. Hvala and S. Strmčnik
Jozef Stefan Institute
Jamova 39, SI-1000 Ljubljana
mario.zec@ijs.si

Abstract. The paper considers the problem of parameter estimation when model parameters are not uniquely identifiable from plant measurements. If uncertainty of parameter estimates is taken into account, also the model results spread over a certain region instead of being a single value. The aim of presented simulation study is to find out whether the estimation of model parameters can be improved in such a way that small enough range of model results is obtained. The results of the study indicate that from measurements it is possible to extract data that is important for the estimation of model parameters relative to a certain model use. By the aid of this data a proper measurement campaign can be specified (*e.g.* proper choice of measured variables, better accuracy of a certain part of measured variable). Simulation study is performed for a simple activated sludge model from wastewater treatment, while the estimation of model parameters is done by Monte Carlo simulation.

Introduction

One of the main obstacles for the use of non-linear biological models (such as standard biological wastewater treatment (WWT) models, like ASM1 [1] or ASM2 [2]) in practice is the estimation of model parameters. Parameters must be fit to the observed process conditions for each WWT plant, because some of the parameters show strong dependency upon plant operation and wastewater composition.

Mathematical models of WWT processes are known as poorly defined systems, which are generally defined as those, for which a valid model structure is not known and the parameter values cannot be accurately specified [6]. Despite of research efforts spent by several research groups, still no established procedure is agreed upon the calibration for this kind of models [7]. Due to identifiability problems unique estimates of non-linear biological model parameters cannot be obtained. It is known that if the model is found unidentifiable, certain parameters can only be estimated in combination with other parameters and different parameter sets can yield almost the same level of performance. Although a good match between model responses and measured variables is obtained, it may mask significant errors in the individual parameters and it could be erroneously concluded that the structure and parameters of the model are appropriate [5].

For example, Jeppsson (1996) has shown that even simple activated sludge model that is theoretically identifiable is practically unidentifiable if a relatively small noise is added to the measured variables. Despite of this fact the methods developed for the estimation of the parameters of well-defined systems are still widely applied in this area. Most often in these cases the parameter estimation procedure results in a unique set of parameter values, and gives a single result when the model is used for a certain purpose. However, it would be more appropriate if the parameter estimation procedure took into account the model uncertainty and identifiability problems, and gave the confidence in estimates. In this way, also the application of the model for a certain purpose would enable to ascertain the confidence in the obtained result. The so obtained result would spread over a certain region instead of being a single value. In case of large confidence limits that indicate unidentifiability of a certain parameter, also large confidence limits of the obtained result are gained.

The above reasoning represents the starting point of the work considering the parameter estimation procedure presented in this paper. The aim of the approach was to take into account the model uncertainty, and to direct the estimation in such a way that a small enough region of model results is obtained when the model is applied for a certain use. A study is aimed at determining how is it possible to choose the measured variables and their measurement policy so that a valid model is obtained for the intended model use. The research has been performed by Monte Carlo simulation and for a simple model of an activated sludge process for WWT.

Methodology

Mathematical model

The parameter estimation procedure will be presented for a simple activated sludge model that describes the bacterial growth and decay in a batch reactor [3]. The model describes the growth of a single organism on a single substrate with no other growth limitations. It is based on Monod equation [4], which is one of the most often used microbial growth-rate models. This equation is the basis for most of the complex mathematical

models of biological WWT processes. The process model used in this paper is the following:

$$\frac{dX}{dt} = \mu_{max} \frac{S}{K_S + S} X - bX \quad ; \quad \frac{dS}{dt} = -\frac{1}{Y} \mu_{max} \frac{S}{K_S + S} X \tag{1}$$

where X represents the biomass concentration [g/m^3], S represents the concentration of substrate [g/m^3], μ_{max} is maximum specific growth rate [h^{-1}], K_S is the substrate half-saturation coefficient [g /m^3], Y is yield factor [(g cell COD formed)/(g COD oxidised)] and b is biomass decay rate [h^{-1}].

To calibrate the model parameters it is necessary to dispose of plant measurements. For the presented simulation study the measurements were obtained by simulating the model with certain parameters values.

In the study, the model was considered to be used for two different purposes. For setting the time when the substrate concentration S drops bellow a certain value (*i.e.* S<5mg/m^3) and for determining the maximum biomass concentration X obtained during the batch.

Parameter estimation

The estimation procedure is based on the agreement between the computed and measured process variable, and is evaluated by an objective function. The objective function is constructed from the following assumptions:

- Around the measured values of process variables it is possible to assign a region. Due to measurement error and model uncertainty it is to expect that the true values of process variables lie within this region.
- The probability density within the region is supposed to be uniform, *i.e.* the probability that a certain value within the region represents the true value of process variable is equal for all the values in the region.
- The model is considered of a good quality if the model response lies within the specified region.

To take into account the above assumptions the objective function based on maximum absolute deviation can be used. It is mathematically expressed in the following way:

$$e_{abs\,max} = e(y_m) = \left| \max_k (c_k) \right| ; \quad c = [c_1, ..., c_k, ..., c_n]^T = U(y_p - y_m) \tag{2}$$

where c denotes a vector of weighted deviations between the measured and computed process output, y_p is a vector of measured values, y_m is a vector of computed values of the process output at corresponding sampling points, n denotes the number of measurements, U is an nxn diagonal matrix with u_{ii} representing the weight of the i-th measured value. Throughout this paper the weighting will be performed in such a way that a certain measurement is included or excluded in the objective function, *i.e.* $u_{ii}=\{0,1\}$.

The chosen parameter values are considered as appropriate if the objective function satisfies the criterion ($e_{absmax} \leq K$). Constant K denotes the maximum allowable deviation between each computed and measured value. Thus K defines a symmetrical region around the measured values inside which the response of models with acceptable parameters values is to be expected. It is important to stress that the choice of K is crucial for the quality of the obtained parameter estimates. It actually expresses an estimate of the model uncertainty and the measurement errors, bellow which the model quality could not be improved.

Definition of different parameter sets and the region of model use

For the estimation of model parameters different parameter sets will be used. Let us first define the vector of model parameters θ, which is defined as $\theta=[\theta_i,...,\theta_k]^T$, where θ_i, $i=1,...,k$, represent individual parameters that need to be estimated, k is the number of parameters. In our case k=4, and $\theta=[X_0, S_0, \mu_{max}, K_S]^T$. A choice of values for X_0, S_0, μ_{max} and K_s defines a parameter vector θ_i in the parameter space. For the parameter estimation, it is reasonable to search for appropriate parameter values only on a subspace Θ, which represents a subspace in the parameter space that includes only reasonable parameter values. A chosen parameter value is considered as reasonable if, considering a given problem, it represents a logical choice of parameter value. In our particular case the process model (1) is based on biological reactions where model parameters have certain physical meanings. Therefore, their reasonable values are already known and are expected to lie in the particular range.

Parameter vectors (PV) θ_i can be classified into different parameter sets. Let us define a set $\mathcal{A}$ as a set of vectors defined in the parameter subspace Θ and a subset $\mathcal{B}$, $\mathcal{B} \subseteq \mathcal{A}$, which includes only those vectors in $\mathcal{A}$ which have acceptable parameters values. Parameter values are considered as acceptable if, when applied in the model, the model response lies in the allowable region around the measured values of the process response. If we define transformation $T:\theta \rightarrow y$ that maps a vector of model parameters θ in the model response y then it holds:

$$T_{\mathcal{A}}: \theta_i \in \mathcal{A} \rightarrow y_i \quad ; \quad y_{mi} = [y_i(t_1), ..., y_i(t_n)]^T \quad ; \quad \text{If } e(y_{mi}) < K \text{ then } \theta_i \in \mathcal{B} \text{ for } \forall\ \theta_i \in \mathcal{A} \tag{3}$$

where y_{mi} is a nx1 vector representing the values of the model response y_i at sampling points t_j, $j=1,...,n$. The above defined relations between the PV and model responses are schematically shown in Fig. 1.

Let us define another transformation $U:\theta \rightarrow z$ that maps a vector of model parameters θ in the model result z ($z \in \Re$), when the model is applied for a certain use. If U is performed on a set of vectors θ_i, then different values z_i are obtained that spread over a region R. This region represents an interval inside which the values of the

model use can be expected. R can be specified by its lower (r_{min}) and upper (r_{max}) bounds and is defined as $R = [r_{min}, r_{max}] = [\min_i(z_i), \max_i(z_i)]$. If U is performed on different model sets also different regions of model use are obtained. In the parameter estimation procedure it is especially interesting to find out whether transformation $U_{\mathcal{B}}$ and the thus obtained $R_{\mathcal{B}}$ give a satisfactory (small enough) region of the model use. This question is considered in the validation process. If the region of model results is not satisfactory, it is necessary to improve the parameter estimation procedure in such a way, that a set of vectors $\mathcal{V}$ with valid parameters values is obtained. In that case $U_{\mathcal{V}}$ produces a valid region of model use $R_{\mathcal{V}}$.

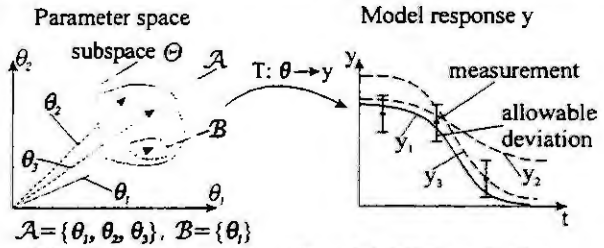


Fig. 1. Transformation of model PV from 2-dim. parameter space into the model response.

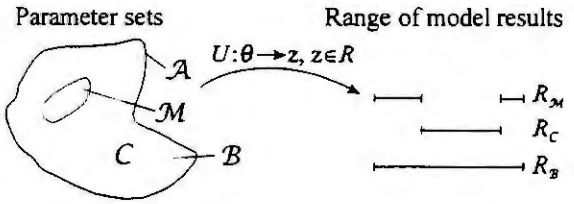


Fig. 2. Relations between different parameter sets and the range of model results.

A possible way of reducing the range of model results (and the parameter space) is by lowering the value of K. Such an approach can be justified only if the model uncertainty and measurement error is improved. In this case a subset C can be defined, $C \subseteq \mathcal{B}$, which is defined according to a reduced value of K_r ($K_r < K$):

$$T_{\mathcal{B}}: \theta_i \in \mathcal{B} \rightarrow y_i \;\; ; \;\; y_{mi} = [y_i(t_1), \ldots, y_i(t_n)]^T \;\; ; \;\; \text{If } e(y_{mi}) < K_r \text{ then } \theta_i \in C \text{ for } \forall\, \theta_i \in \mathcal{B} \qquad (4)$$

For C it is to expect that U_C produces a narrower range of model results then $U_{\mathcal{B}}$, and for some PV in C', $C' = \mathcal{B} \setminus C$, transformation U produces results inside $R_{\mathcal{B}}$ but outside R_C. Parameter vectors with the above properties can be denoted as a set $\mathcal{M}$, which is a subset of C', $\mathcal{M} \subseteq C'$. In the parameter estimation procedure PV in $\mathcal{M}$ should be excluded, so that narrower and thus valid range of model use is obtained. The relations between the parameter sets and the range of results are graphically shown in Fig. 2.

Considering the so defined $\mathcal{M}$, it is interesting to look at the model response characteristics of the PV in $\mathcal{M}$ obtained by $T_{\mathcal{M}}$. The hypothesis is that they include some important features that can be usefully applied in the parameter estimation. With this information, measurements can be designed in such way that estimates of vectors with valid parameters values $\mathcal{V}$ are obtained, so that they are appropriate for a certain model use.

Results

Model and simulations for the assessment of measurements and weights in the objective function for the desired purpose of the model use were performed in Matlab-Simulink environment. Simulations and simulated measurements were performed from t=0h to t=30h, but figures shows only the most interesting part from t=0h to t=20h. The subspace Θ is defined according to the reasonable parameters ranges (X_0 =2 to 6 g/m^3, S_0 =20 to 60 g/m^3, μ_{max}=0.15 to 0.36 h^{-1} and K_s=5 to 55 g/m^3) obtained from the literature. The constant K for the determination of subset $\mathcal{B}$ is set to K=2.76, and K_r=2.208 for the determination of C.

The left diagram in Fig. 3 shows model responses for parameter values from $\mathcal{B}$. It can be observed that responses completely fill the region defined by K. Black middle line represents measured response of the process, outer black lines show maximum allowed deviation between each computed and measured value for K=2.76. The middle diagram in Fig. 3 shows responses for parameter vectors in $\mathcal{M}$ and the model use "time". In this case responses in the first part fill the whole allowed deviation range between computed and measured value. But from the time t=9h they split into two groups that lie near the bounds of the maximum allowed deviation. If transformation $U_{\mathcal{M}}$ is used, then it maps vectors from one of these groups near to the lower bound of the results range (left interval of the $R_{\mathcal{M}}$ in Fig. 2) and the other group near to the upper bound of the results range (right interval of the $R_{\mathcal{M}}$ in Fig. 2).

Similar conclusions can be derived from the right diagram in Fig. 3 that shows model responses for parameter vectors in $\mathcal{M}$ and for the model use "maxX". The only difference is that responses group in the first part of the measured response. From the observed characteristics it can be concluded that in this case for the definition of the results range the most important part of the model response is the one where responses of the PV from subset $\mathcal{M}$ split into two groups.

The above conclusions were confirmed by simulation experiments. In these simulations only that part of the measurements, that is considered to be "important" (where responses of vectors from subset $\mathcal{M}$ form two groups) for the desired model purpose is used for the definition of subset $\mathcal{B}$. For that reason values of the diagonal elements of matrix U were set to 1 (for that part of the measurements that are considered to be "important") and 0 (for the part of measurements that are not used within the objective function).

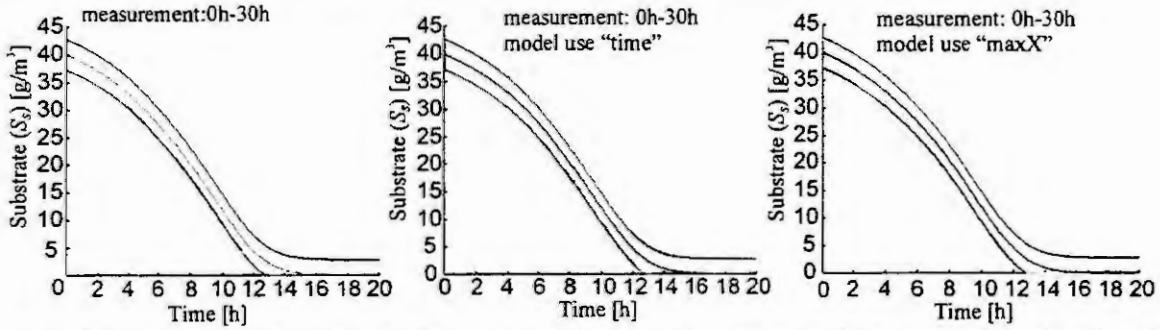


Fig. 3: Left-model response based on the subset $\mathcal{B}$, middle and right-responses of the vectors from the subset $\mathcal{M}$.

If the characteristics from Fig. 3 are taken into account, then measuring of the $S_S(t)$ can be reduced. For the model use "time" $S_S(t)$ can be measured only from t=9h to t=20h, and for the model use "maxX" $S_S(t)$ can be measured only from t=0h to t=9h. Results of the simulations with a reduced number of measurements are shown in Table 1.

Table 1 Results range when different sets of model PV based on measurement $S_S(t)$ are used.

Time of measurement		0h-30h	9h-20h	0h-9h
Model use "time"	r_{min}	11.10	11.10	10,40
	r_{max}	13.05	13.05	16,25
Model use "maxX"	r_{min}	22.49	15.39	22,29
	r_{max}	29.25	39.17	29,08

Conclusions

Comparison of the results ranges when only one part of measurements is used for the determination of $\mathcal{B}$ with ranges when the whole set of measurements is used shows that, in spite of the reduced measurements, very similar results ranges are gained (shaded parts of the Table 1). It can be concluded, that if the measurements are planned in advance, or additional measurements can be performed, it is reasonable to put more efforts in accurate measurement of this "important" part of the measured variable. With more accurate measurement of this part of the measurements narrower results range (and more appropriate parameter space) can be gained. Responses of the vectors from $\mathcal{M}$ can provide information about this "important" part of the measurements.

References

1. Henze, M., Grady Jr, C. P. L., Gujer, W., Marais, G. v. R., Matsuo, T., Activated Sludge Model No. 1. IAWPRC, London, 1987.
2. Henze, M., Gujer, W., Mino, T., Matsuo, T., Wentzel, M. C., Marais, G. v. R., Activated Sludge Model No. 2. IAWPRC, London, 1995.
3. Jeppsson, U., Modelling Aspects of Wastewater Treatment Processes. Ph D Thesis, Printed by Reprocentralen, Lund University, Lund, 1996.
4. Monod, J., The growth of bacterial cultures. Ann. Rev. Microbiol., 3 (1949), 371-394.
5. Murray-Smith, D.J., Methods for the external validation of continuous system simulation models: a review. Mathematical and Computer Modelling of Dynamical Systems, 4(1) (1998), 5-31.
6. Sperling, M. v., Parameter estimation and sensitivity analysis of an activated sludge model using monte carlo simulation and the analyst's involvement. Water Sci. Tech., 28(11-12) (1993), 219-229.
7. Weijers, S.R., Preisig, H.A., Buunen, A., Wouda, T. W. M., Parameter estimation of activated sludge model no. 1 from full scale plant input/output data. In: Proc. of the European Control Conference ECC'97, Brussels, 1997.

NONLINEAR MODEL REDUCTION – METHOD AND CAE-TOOL DEVELOPMENT

M. Kordt and J. Ackermann
DLR (German Aerospace Research Center),
Institute for Robotics and Systemdynamics, Oberpfaffenhofen, D–82234 Weßling

Abstract. New methods for nonlinear model reduction of dynamic models, described by nonlinear differential equations have been developed. The methods are based on singular perturbations, weak coupling and cost functionals for computing and analyzing the reduced order model. By introducing a formally affine nonlinear model structure, the cost functional vector optimization yields closed expressions for the reduced order system. Via *Lagrange* multipliers even constraints with regard to the reduced order system can be considered, only extending these closed expressions. This leads to the notion of smaller and control–oriented realizations. The methods have been programmed and integrated into a MATLAB based Toolbox, termed NEON (NonlinEar Order reductioN). NEON uses standard MATLAB/SIMULINK and the GUI, OPTIMIZATION, CONTROL and SYMBOLIC Toolboxes. Via NEON the methods are here applied to a structural dynamic aircraft model, in order to achieve a very low order model, suited for integral structural dynamic and flight mechanical controller design.

1. Introduction

Given a nonlinear vector differential equation of order n, nonlinear order reduction methods compute a differential equation for the dominant quantities of order $\tilde{n} < n$, such that the resulting trajectories $\tilde{\mathbf{x}}(t)$ are as close as possible to those of the original system. Only a few nonlinear order methods are presently available.

The method by *Hasenjäger* [6] applies a linear order reduction method to the linear part of the model and adds the nonlinear part of the differential equations for the dominant quantities unchanged to the reduced order model. The method by *Lohmann* [15, 16] overcomes this rough approximation. It considers a set of representative simulations of the original system, covering its operating domain densely. By considering two independent cost functionals for these simulations the reduced order system can be computed analytically. Minimization of the first cost functional yields a constant matrix $\mathbf{W}$ which reconstructs an approximation $\mathbf{x}_{approx}(t)$ of the full trajectory $\mathbf{x}(t)$ out of the trajectory of the reduced system: $\mathbf{x}_{approx}(t) = \mathbf{W}\tilde{\mathbf{x}}(t)$ such that $\mathbf{x}_{approx}(t) \cong \mathbf{x}(t)$, $\forall t \in \mathbb{R}_{\geq 0}$. Thereby, the nonlinearities of the original system can be transferred to the reduced order system. The second cost functional computes the reduced order system, in particular new weighting factors for the nonlinearities. However, in general every nonlinearity of the original system enters in *every* differential equation of the reduced order system. This is harmful for nonlinear controller design [8, 11, 20], understanding of the main dynamic effects and real–time applications. Therefore the notion of model reduction is here introduced which simultaneously does order reduction and simplification of the nonlinearities. It is achieved by a cost functional vector formulation including equality and inequality constraints.

The singular perturbation reduction method [10] allows to compute the reduced order system analytically and without simulations, if the non–dominant dynamics is high frequent and weakly excited. It is here combined with a model reduction for weakly coupled systems. Moreover, a simple applicability criterion in terms of a scalar measure is here derived which also identifies suitable combinations of weakly excited, high frequent and weakly coupled dynamics.

Concerning the application of order reduction method to structural dynamic aircraft models so far only the method of *Hasenjäger* has been applied in combination with linear structural dynamic model reduction techniques [7, 9, 12, 17]. However, there is an increasing interest to consider nonlinear structural dynamic aircraft models for controller design [1]. Due to improvements in nonlinear control this holds for many other applications, [11, 20, 8].

2. Cost Functional Based Nonlinear Model Reduction

Given the original system of nonlinear differential equations of order n in $\mathbb{R}^n$: $\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))$, $\mathbf{x}(0) = \mathbf{x}_0$, $t \in \mathbb{R}_{\geq 0}$ with input vector function $\mathbf{u}$, $\mathbf{u}(t) \in \mathbb{R}^m$, $\forall t \in \mathbb{R}_{\geq 0}$. It is formulated in formally affine representation: $\dot{\mathbf{x}} = \mathbf{E}\ \mathbf{m}$, where: $\mathbf{E} = [\mathbf{A}\ \mathbf{B}\ \mathbf{F}]$ is a constant matrix , $\mathbf{m} = [\mathbf{x}\ \mathbf{u}\ \mathbf{g}(\mathbf{x}, \mathbf{u})]^T$, i.e. the system is not only affine in the input $\mathbf{u}(t)$ but also in $\mathbf{g}(\mathbf{x}(t), \mathbf{u}(t))$ at arbitrarily fixed time t. $\mathbf{g}(\mathbf{x}(t), \mathbf{u}(t))$ is a tall vector containing all nonlinearities of the system up to a constant factor.

First, the dominant state vector $\mathbf{x}_d$ of dimension $\tilde{n} < n$ has to be chosen out of the state vector $\mathbf{x}$. Thereby the dimension of the reduced order system is fixed. The dominant state vector contains the states which describe the dominant dynamics and whose evolution in time should be described by differential equations. The dominant states are e.g. measured, controlled or commanded states or states entering in the significant nonlinearities. In case they are not obvious a cost functional based dominance analysis [15, 16] is suggested.

The core of the reduction methods are cost functionals, constituting a cost functional vector $\mathbf{j} = [j_1, ..., j_N]^T$ and estimating the deviations between the dynamics of the original and the reduced order model. The cost functionals are evaluated for a finite number of input signals, representing the operating domain of the system. Generation of these signals is supported by a signal editor, figure 1. The cost functionals are minimized with regard to the parameters of the reduced order system under constraints, concerning the physical and structural properties of the reduced order system. Such properties are (i) linear and nonlinear coupling or decoupling properties between subsystems, (ii) steady state accuracy of the reduced order model, (iii) the number of nonlinearities or (iv) the contribution of the linear terms and the nonlinear terms within the differential equations. These constraints are particularly suited to keep only the most relevant nonlinearities in the system. The method is iterative because the reduced order system is computed based on representative signals: In a detailed assessment the reduced system has to be simulated for different input signals, covering the whole operating domain densely. The cost functionals can be formulated for each individual test signal, e.g. as quadratic cost functionals for a good time-averaged approximation of the dominant states:

$$j_{1,i} = \int_0^{T_i} q^2(t) ||\dot{\mathbf{x}}_d - \tilde{\mathbf{E}}\,\mathbf{m}_d||^2 dt \stackrel{!}{=} min(\tilde{\mathbf{E}}), \tag{1}$$

where $\mathbf{m}_d = [\mathbf{x}_d \quad \mathbf{u} \quad \mathbf{g}(\mathbf{W}\,\mathbf{x}_d, \mathbf{u})]^T$. $\tilde{\mathbf{E}} = [\tilde{\mathbf{A}} \quad \tilde{\mathbf{B}} \quad \tilde{\mathbf{F}}]$ includes the matrices of the reduced order system, $q^2(t)$ is a positive function for time–weighting, $||.||$ is the standard Euclidean norm. The matrix $\mathbf{W}$ reconstructs the full state $\mathbf{x}$ out of $\mathbf{x}_d$, so that the nonlinearities can be transferred unchanged to the reduced order system. It can be computed via a second quadratic cost functional:

$$j_{2,i} = \int_0^{T_i} q_{\mathbf{W}}^2(t) ||\mathbf{x} - \mathbf{W}\,\mathbf{x}_d||^2 dt \stackrel{!}{=} min(\mathbf{W}). \tag{2}$$

$q_{\mathbf{W}}^2(t)$ is a positive function for time–weighting. The quadratic cost functional approach allows to cover several signals $i = 1, ..., N$ with corresponding finite simulation time T_i by one cost functional:

$$j_1 = \int_0^{T_1+...+T_N} q^2(t) ||\dot{\mathbf{x}}_d - \tilde{\mathbf{E}}\,\mathbf{m}_d||^2 dt = trace\left\{(\dot{\mathbf{X}}_d - \tilde{\mathbf{E}}\,\mathbf{M}_d)\,\mathbf{Q}\,\mathbf{Q}^T\,(\dot{\mathbf{X}}_d - \tilde{\mathbf{E}}\,\mathbf{M}_d)^T\right\} \stackrel{!}{=} min(\tilde{\mathbf{E}}), \tag{3}$$

$$j_2 = trace\{(\mathbf{X} - \mathbf{W}\,\mathbf{X}_d)\mathbf{Q}_{\mathbf{W}}\,\mathbf{Q}_{\mathbf{W}}^T(\mathbf{X} - \mathbf{W}\,\mathbf{X}_d)^T\} \stackrel{!}{=} min(\mathbf{W}), \tag{4}$$

where the cost functionals have been discretely evaluated:
$\mathbf{M}_d = [\mathbf{m}_d(t_{11}), \cdots, \mathbf{m}_d(t_{1,n_1} = T_1), \cdots, \mathbf{m}_d(t_{N,1}), \cdots, \mathbf{m}_d(t_{N,n_N} = T_N)]$, i.e. the column vectors $\mathbf{m}_d$, evaluated for each discrete time step and each simulation, are concatenated in a fat matrix, equivalently: $\dot{\mathbf{X}}_d = [\dot{\mathbf{x}}_d(t_{11}), \cdots, \dot{\mathbf{x}}_d(t_{N,n_N} = T_N)]$, $\mathbf{X}_d = [\mathbf{x}_d(t_{11}), \cdots, \mathbf{x}_d(t_{N,n_N} = T_N)]$ and
$\mathbf{Q} = diag[q(t_{11}), ..., q(t_{1,n_1}), ..., q(t_{N,1}), ..., q(t_{N,n_N})]$, $\mathbf{Q}_W = diag[q_{\mathbf{W}}(t_{11}), ..., q_{\mathbf{W}}(t_{N,n_N})]$. According to this simulation based formulation of the cost functionals, not only closed nonlinear functions, but also nonlinearities given as data arrays and hysteresis can be considered in $\mathbf{g}(\mathbf{x}, \mathbf{u})$. The cost functional vector optimization problem eq. (3) and (4) is solved in two steps: Firstly, the reconstruction matrix $\mathbf{W}$ is analytically computed from the minimum condition corresponding to eq. (4). Secondly, by substituting this matrix $\mathbf{W}$ into $\mathbf{M}_d$ the optimization problem for the parameters of the reduced order system $\tilde{\mathbf{E}}$ is analytically solved from the minimum condition corresponding to eq. (3), yielding a closed expression for the system matrices of the reduced order system:

$$\tilde{\mathbf{E}} \quad = \quad \dot{\mathbf{X}}_d\,\mathbf{Q}\,\mathbf{Q}^T\,\mathbf{M}_d^T\,(\mathbf{M}_d\,\mathbf{Q}\,\mathbf{Q}^T\,\mathbf{M}_d^T)^{-1}. \tag{5}$$

Via *Lagrange* multipliers, constraints of the type $\mathbf{G}\,\tilde{\mathbf{E}}\,\mathbf{H} - \mathbf{L} = 0$ can be included in the cost functional eq. (3), still allowing to compute the reduced order system via a closed expression:

$$\tilde{\mathbf{E}} = \tilde{\mathbf{E}}_R + \left[\mathbf{L} - \tilde{\mathbf{E}}_R\,\mathbf{H}\right]\,[\Omega\,\mathbf{H}]^{-1}\,\Omega, \tag{6}$$

where: $\Omega = \mathbf{H}^T (\mathbf{M}_d \mathbf{Q} \mathbf{Q}^T \mathbf{M}_d^T)^{-1}$ and: $\tilde{\mathbf{E}}_R = \dot{\mathbf{X}}_d \mathbf{Q} \mathbf{Q}^T \mathbf{M}_d^T (\mathbf{M}_d \mathbf{Q} \mathbf{Q}^T \mathbf{M}_d^T)^{-1}$. (7)

For brevity, G has here been chosen as unity matrix. In general, it only has to have maximal row rank[1], H has to have maximal column rank. G and H select the desired elements from $\tilde{\mathbf{E}}$. The matrix L contains the desired values of these elements. Hereby, the structural and physical properties (i) to (iv) can be imposed on the reduced order system. The approach is open to arbitrary cost functionals, which not necessarily need to be analytically solvable. The only consequence is that an iterative optimization becomes necessary. An example is a maximum norm cost functional

$$j_i = max_{t \in [0,T_i]} ||\dot{\mathbf{x}}_d - \tilde{\mathbf{E}} \mathbf{m}_d|| \stackrel{!}{=} min(\tilde{\mathbf{E}}, \mathbf{W}), \quad (8)$$

e.g. to cover peak phenomena exactly. Remarkably $\tilde{\mathbf{E}}$ and W can be computed simultaneously, ensuring a more precise computation of the reduced order system. Within such an iterative optimization, more general inequality constraints $|f_{c,i}(\tilde{\mathbf{E}})| \leq const.$, $i = 1, ..., N_c$ can be used.

The setup of this section allows to achieve a so-called smaller or a so-called control-oriented realization, which are defined in the following and which are of high practical interest in nonlinear control system design and understanding of nonlinear systems.

Definition (*Smaller Realization, Control-Oriented Realization*):
A system $\sum_\alpha : \dot{\mathbf{x}}_\alpha = \mathbf{A}_\alpha \mathbf{x}_\alpha + \mathbf{B}_\alpha \mathbf{u}_\alpha + \mathbf{F}_\alpha \mathbf{g}_\alpha(\mathbf{W}_\alpha \mathbf{x}_\alpha, \mathbf{u}_\alpha)$ has the *nonlinearity index* N_α, if and only if, it has N_α non-vanishing entries in the matrix $\mathbf{F}_\alpha$.
The system $\sum_\beta : \dot{\mathbf{x}}_\beta = \mathbf{A}_\beta \mathbf{x}_\beta + \mathbf{B}_\beta \mathbf{u}_\beta + \mathbf{F}_\beta \mathbf{g}_\beta(\mathbf{W}_\beta \mathbf{x}_\beta, \mathbf{u}_\beta)$ is termed *smaller realization* w.r.t. to the system $\sum_\alpha$, if and only if:
(a) it has a smaller order: $n_\beta < n_\alpha$, (b) the nonlinearity index does not increase: $N_\beta \leq N_\alpha$.
The system $\sum_\beta$ is termed *control-oriented realization* w.r.t. the system $\sum_\alpha$, if and only if: (a) the order of the system $\sum_\beta$ does not increase: $n_\beta \leq n_\alpha$, (b) the matrices $\mathbf{A}_\beta$, $\mathbf{B}_\beta$, $\mathbf{F}_\beta$, $\mathbf{W}_\beta$ match predefined properties required for controller design. □

Such *smaller* or *control-oriented realizations* in particular correspond to zero elements in the matrix $\tilde{\mathbf{F}}$. They can be achieved by requiring $\mathbf{G}\ \tilde{\mathbf{E}}\ \mathbf{H} = \mathbf{0}$ (constraints for analytically closed optimization) or $f_c(\tilde{\mathbf{F}}) = \tilde{F}_{i_1 j_1}^2 + \tilde{F}_{i_2 j_2}^2 + ... < \epsilon$, where ϵ is a small number. $\tilde{F}_{i_1 j_1}$ are certain elements chosen out of $\tilde{\mathbf{F}}$. Instead of the Euclidean norm, a wide range of inequality formulations according to optimization theory can be used for the iterative optimization. *Control-oriented realizations* may require not only constraints w.r.t. $\tilde{\mathbf{F}}$, but also w.r.t. $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{B}}$. The essential advantage for controller design is that a synthesis model structure identification can be done, i.e. the question is answered, whether an approximate model of the given system matches a model structure of a classified system, suited for a particular nonlinear controller design method like [8, 5, 11, 20]. In contrast to a minimal realization [8] which is very hard to construct, for achieving smaller realizations, now, constructive methods have here been developed.

3. Nonlinear order reduction by compensation

The second reduction method, termed *reduction by compensation*, combines singular perturbation [10] and a reduction method for systems with weakly coupled subsystems. In contrast to the cost *functional based reduction* the reduced system can be computed analytically straight from the differential equations of the original system. Thereby, iterations in consequence of inadequate test signal sets are avoided. To derive the method it is assumed that:
(A1) the system can be partitioned in a dominant and a non-dominant part *d* and *nd*, where only the dominant states are to be retained in the reduced order system:

$$\dot{\mathbf{x}}_d = \mathbf{A}_{11} \mathbf{x}_d + \mathbf{A}_{12} \mathbf{x}_{nd} + \mathbf{B}_1 \mathbf{u} + \mathbf{F}_1 \mathbf{g}(\mathbf{x}_d, \mathbf{u}), \quad (9)$$

$$\dot{\mathbf{x}}_{nd} = \mathbf{A}_{21} \mathbf{x}_d + \mathbf{A}_{22} \mathbf{x}_{nd} + \mathbf{B}_2 \mathbf{u} + \mathbf{F}_2 \mathbf{g}(\mathbf{x}_d, \mathbf{u}), \quad (10)$$

(A2) the vector $\mathbf{g}(\mathbf{x}_d, \mathbf{u})$ only contains dominant states. In case this is not given, the matrix W, c.f. eq. (4), can be used to achieve it approximately. This approximation has to be thoroughly analyzed in state space in case of transition layers [3], i.e. the problem of nonunique solutions of the equation $\mathbf{A}_{21} \mathbf{x}_d + \mathbf{A}_{22} \mathbf{x}_{nd} + \mathbf{B}_2 \mathbf{u} + \mathbf{F}_2 \mathbf{g}(\mathbf{x}_d, \mathbf{u}) = \mathbf{0}$ with regard to $\mathbf{x}_{nd}$. A particular advantage of this 2nd assumption is that $\mathbf{g}(\mathbf{x}_d, \mathbf{u})$ is not required to consist of analytically closed functions, but allows nonlinearities to

[1] To solve the optimization problem in this case requires Kronecker-product and yields more lengthy closed expression.

be given in terms of data arrays. Moreover, hysteresis can be considered. Concerning assumption (A1) the nonlinear dominance analysis developed by *Lohmann* [15, 16] or a criterion, which is derived in the following, can be used.

In case of singular perturbations, $\dot{\mathbf{x}}_{nd}$ is assumed to be small, i.e. the corresponding dynamic is weakly excited and high frequent, e.g. due to small time constants, small masses or large gains, [10]. If $\mathbf{A}_{22}$ is regular, the remaining algebraic equation

$$0 = \mathbf{A}_{21}\mathbf{x}_d + \mathbf{A}_{22}\mathbf{x}_{nd} + \mathbf{B}_2\mathbf{u} + \mathbf{F}_2\mathbf{g}(\mathbf{x}_d, \mathbf{u}) \tag{11}$$

serves to eliminate $\mathbf{x}_{nd}$ in eq. (9):

$$\dot{\mathbf{x}}_d = (\mathbf{A}_{11} - \mathbf{A}_{12}\,\mathbf{A}_{22}^{-1}\,\mathbf{A}_{21})\,\mathbf{x}_d + (\mathbf{B}_1 - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{B}_2)\mathbf{u} + (\mathbf{F}_1 - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{F}_2)\mathbf{g}(\mathbf{x}_d, \mathbf{u}). \tag{12}$$

Next, a reduction technique for weakly coupled systems is presented. In weakly coupled systems all elements in the matrix $\mathbf{A}_{12}$ are small, i.e. $|\mathbf{A}_{12}\,\mathbf{x}_{nd}(t)| \ll |\dot{\mathbf{x}}_d(t)|$, $\forall t \in \mathbb{R}_{\geq 0}$ and for all relevant initial conditions $\mathbf{x}_0$ and input signals $\mathbf{u}$. Via this condition the case of different amplitude scaling of $\mathbf{x}_d$ and $\mathbf{x}_{nd}$ is covered. This condition allows to neglect the transient dynamic effect of eq. (10) in eq. (9), if the time scales of the non-dominant systems are of the same order or smaller and in particular, if $\dot{\mathbf{x}}_{nd}$ is small. Consequently, the reduced order system can be computed according to eq. (12). In order to combine both methods and simultaneously answer the question of their applicability, the assumption of small $\dot{\mathbf{x}}_{nd}$ is withdrawn: Substituting then eq. (10) in eq. (9) yields:

$$\begin{aligned} \dot{\mathbf{x}}_d - \mathbf{A}_{12}\,\mathbf{A}_{22}^{-1}\dot{\mathbf{x}}_{nd} &= (\mathbf{A}_{11} - \mathbf{A}_{12}\,\mathbf{A}_{22}^{-1}\,\mathbf{A}_{21})\,\mathbf{x}_d + (\mathbf{B}_1 - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{B}_2)\mathbf{u} \\ &\quad + (\mathbf{F}_1 - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{F}_2)\mathbf{g}(\mathbf{x}_d, \mathbf{u}), \end{aligned} \tag{13}$$

$$\dot{\mathbf{x}}_{nd} = \mathbf{A}_{21}\,\mathbf{x}_d + \mathbf{A}_{22}\mathbf{x}_{nd} + \mathbf{B}_2\,\mathbf{u} + \mathbf{F}_2\,\mathbf{g}(\mathbf{x}_d, \mathbf{u}). \tag{14}$$

Now, from the first equation all conditions can be read, when a combined reduction, based on singular perturbation and weak coupling applies: (i) All coupling constants from the non–dominant to the dominant system are small (special case of reduction for weakly coupled system). We emphasize that the case of same orders of time scales (or frequencies in the special case of linear systems) in the retained and the omitted dynamics can be tackled. (ii) $\mathbf{A}_{22}^{-1}$ or $\dot{\mathbf{x}}_{nd}$ is small (special case of singular perturbation). (iii) Both special cases interfere adequately and cause the vector $\mathbf{A}_{12}\,\mathbf{A}_{22}^{-1}\,\dot{\mathbf{x}}_{nd}$ to be small. This interference allows condition (i) and (ii) to be violated to some extent, but in a complementary fashion: This is the essential case to which the applicability of both methods is extended.

In all three cases, the reduced order system can be computed as in case of singular perturbations, eq. (12). The method is termed *reduction by compensation* because in the reduced order system the above listed effects are all compensated by additive corrections to the parameters of the vector differential equation of the retained states. Concerning the nonlinear part of the reduced order system, additional nonlinearities are generated by the term $\mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{F}_2$. It ensures, that there are no extra nonlinearities in the reduced order system, if the non–dominant dynamics is linear. In case of a physical model the weighting factors of the extra nonlinearities consist of physical parameters. The significance of the extra nonlinearities can then be analyzed physically. The most essential consequence of eq. (13) is, that it suggests a cost functional to estimate, if one of the conditions (i) to (iii) is fulfilled. Thereby assumption (A1) can be tested.

Remark (Applicability criterion for the reduction by compensation):
The system (9), (10) can be reduced to eq. (12), if

$$J = \frac{trace\{\mathbf{A}_{12}\,\mathbf{A}_{22}^{-1}\,\dot{\mathbf{X}}_{nd}\,\dot{\mathbf{X}}_{nd}^T(\mathbf{A}_{12}\,\mathbf{A}_{22}^{-1})^T\}}{trace\{\dot{\mathbf{X}}_d\,\dot{\mathbf{X}}_d^T\}} \ll 1, \tag{15}$$

i.e. the relative time-averaged contribution of $\mathbf{A}_{12}\,\mathbf{A}_{22}^{-1}\,\dot{\mathbf{x}}_{nd}$ to $\dot{\mathbf{x}}_d$ is small. □

Notice that the cost functional (15) only requires simulation data of the original system and a suggestion of the dominant states. The proof is obvious: Introducing the time integral yields

$$J = \frac{trace\left\{\int_0^\infty(\mathbf{A}_{12}\,\mathbf{A}_{22}^{-1}\,\dot{\mathbf{x}}_{nd})\,(\mathbf{A}_{12}\,\mathbf{A}_{22}^{-1}\,\dot{\mathbf{x}}_{nd})^T dt\right\}}{trace\left\{\int_0^\infty \dot{\mathbf{x}}_d\,\dot{\mathbf{x}}_d{}^T dt\right\}} \ll 1, \tag{16}$$

i.e. the relative quadratic time–averaged contribution of $\mathbf{A}_{12}\mathbf{A}_{22}^{-1}\dot{\mathbf{x}}_{nd}$ to $\dot{\mathbf{x}}_d$ is small. An essential advantage of the criterion is that it is a matrix criterion, i.e. not all entries of the matrices $\mathbf{A}_{12}$ and $\mathbf{A}_{22}^{-1}$ have to be analyzed individually, whether they constitute small constants for the whole operating domain. This is done via the criterion in one shot, i.e. simultaneously for all matrix entries and thereby for all coupling and time constants under consideration. In this sense, the criterion also supports and confirms the choice or the definition of small constants within singular perturbation, [11].

Via linearization $\dot{\mathbf{x}}_i = \mathbf{A}_i\ \mathbf{x}_i + \mathbf{B}_i\ \mathbf{u}$ of the system at a sufficient dense set of operating points i, the cost functional can even be evaluated without simulations. The index i is dropped in the following. The corresponding linearized system matrices ($\mathbf{A}$, $\mathbf{B}$) and the evaluation of the cost functional can be automatically done within NEON, using SYMBOLIC TOOLBOX: The idea is to consider a sufficient dense set of initial conditions $\mathbf{x}_0$ and step inputs of height $\mathbf{u}_0$ for each operating point i. Via partitioning $\mathbf{x}$ in $\mathbf{x}_d = [\mathbf{I}_d \quad 0]\mathbf{x}$ and $\mathbf{x}_{nd} = [0 \quad \mathbf{I}_{nd}]\mathbf{x}$, where $\mathbf{I}_d$ and $\mathbf{I}_{nd}$ are unity matrices of dimension $\tilde{n}$ and $n - \tilde{n}$, the integrals in the numerator and the denominator can be evaluated via the integral:

$$\int_0^\infty \dot{\mathbf{x}}\dot{\mathbf{x}}^T dt = \int_0^\infty (\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u})(\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u})^T dt\,. \tag{17}$$

Now, solving the linearized differential equation at an arbitrary operating point i for a step input of height $\mathbf{u}_0$ and the initial condition $\mathbf{x}_0$, allows to evaluate eq. (17) without simulation:

$$\int_0^\infty \dot{\mathbf{x}}\dot{\mathbf{x}}^T dt = \mathbf{A}\mathbf{S}_{\mathbf{x}_0,\mathbf{x}_0}\mathbf{A}^T + \mathbf{A}\mathbf{S}_{\mathbf{x}_0,\mathbf{B}\mathbf{u}_0} + \mathbf{S}_{\mathbf{B}\mathbf{u}_0,\mathbf{x}_0}\mathbf{A}^T + \mathbf{S}_{\mathbf{B}\mathbf{u}_0,\mathbf{B}\mathbf{u}_0}\,,$$
$$\text{where: } \mathbf{S}_{\mathbf{B}\mathbf{u}_0,\mathbf{B}\mathbf{u}_0} = \int_0^\infty \mathrm{e}^{\mathbf{A}t}\mathbf{B}\mathbf{u}_0(\mathbf{B}\mathbf{u}_0)^T(\mathrm{e}^{\mathbf{A}t})^T dt.$$

The other integrals are abbreviated similarly: $\mathbf{S}_{\mathbf{x}_0,\mathbf{x}_0}$, $\mathbf{S}_{\mathbf{x}_0,\mathbf{B}\mathbf{u}_0}$, $\mathbf{S}_{\mathbf{B}\mathbf{u}_0,\mathbf{x}_0}$. These four integrals $\mathbf{S}_{*,\bullet}$ can be evaluated using the *Lyapunov* equation:

$$\mathbf{A}\,\mathbf{S}_{*,\bullet} + \mathbf{S}_{*,\bullet}\,\mathbf{A}^T \;=\; -\mathbf{W}_{*,\bullet}\,, \tag{18}$$

if and only if all eigenvalues of $\mathbf{A}$ are stable, where $\mathbf{W}_{*,\bullet}$ is defined in analogy to $\mathbf{S}_{*,\bullet}$, e.g. $\mathbf{W}_{\mathbf{B}\mathbf{u}_0,\mathbf{B}\mathbf{u}_0} = \mathbf{B}\,\mathbf{u}_0\,\mathbf{u}_0{}^T\,\mathbf{B}^T$. Thereby, the cost functional can be written as:

$$J = \frac{trace\left\{\mathbf{A}_{12}\mathbf{A}_{22}^{-1}\begin{bmatrix} 0 & \mathbf{I}_{nd}\end{bmatrix}\Sigma\begin{bmatrix} 0 \\ \mathbf{I}_{nd}\end{bmatrix}(\mathbf{A}_{12}\mathbf{A}_{22}^{-1})^T\right\}}{trace\left\{\begin{bmatrix}\mathbf{I}_d & 0\end{bmatrix}\Sigma\begin{bmatrix}\mathbf{I}_d \\ 0\end{bmatrix}\right\}},$$

$$\Sigma = (\mathbf{A}\mathbf{S}_{\mathbf{x}_0,\mathbf{x}_0}\mathbf{A}^T + \mathbf{A}\mathbf{S}_{\mathbf{x}_0,\mathbf{B}\mathbf{u}_0} + \mathbf{S}_{\mathbf{B}\mathbf{u}_0,\mathbf{x}_0}\mathbf{A}^T + \mathbf{S}_{\mathbf{B}\mathbf{u}_0,\mathbf{B}\mathbf{u}_0}).$$

It only contains the chosen vectors $\mathbf{x}_0$, $\mathbf{u}_0$ and the system matrices, so that it can be evaluated very efficiently, compared to a coupling analysis based on simulations of the two coupled subsystem. The consistency of the formula is obvious from the quadratic term in $\mathbf{A}_{12}\ \mathbf{A}_{22}^{-1}$ in the numerator.

4. Architecture of the MATLAB based Toolbox NEON

The above methods are completely implemented in the MATLAB based Toolbox NEON. The Toolbox is started by typing NEON at the MATLAB–prompt. A graphical user interface window appears. From here reduced order systems can be iteratively computed, comparatively analyzed and detailed assessments can be made via pulldown and selection menus consisting of several text fields or yes/no decision, buttons, input dialog windows and online-help, cf. figure 1. Consequently, the engineer interacts with NEON in terms of pure mathematics. The open MATLAB environment allows him to pre- and post-process data and results easily. The data of the model reduction process are automatically stored in a hierarchical data structure with various data types. The data structure is realized as a structure array and hierarchically visualized in a listbox, figure 1.

5. Example: Model reduction of a structural dynamic aircraft model via NEON for derivation of a control–oriented realization for structural dynamic controller design

Structural dynamic modeling considers both the rigid body and the elastic motion of an aircraft. The rigid body motion in six degrees of freedom (longitudinal and lateral motion) is described by the three

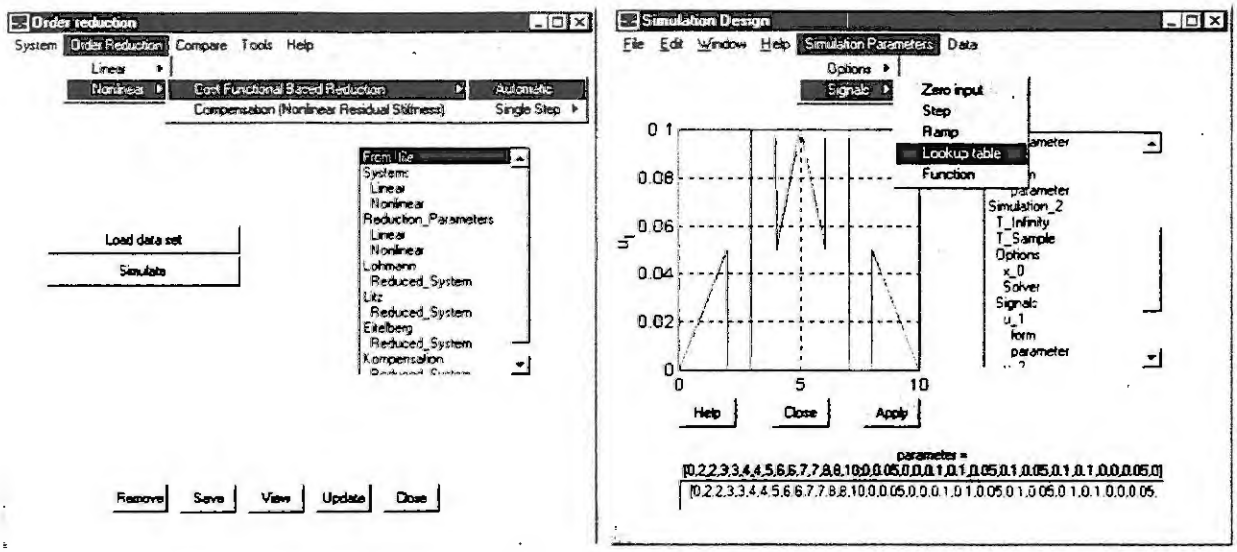


Figure 1: Left: Architecture of the Toolbox NEON. Right: Signal Editor, graphical user interface for defining sets of input signals for computation and analysis of a reduced order system.

velocity components u_B, v_B, w_B and the three angular velocity components p_B, q_B, r_B with regard to a body-fixed frame B and the three Euler angles ϕ, θ, ψ, [17]. Newton and Euler equation of motion and the relationship between the Euler angles and the angular velocities [17] yield the vector differential equation

$$\dot{\mathbf{x}}_R = \mathbf{A}_{11}\mathbf{x}_R + \mathbf{B}_1\mathbf{c} + \mathbf{F}\mathbf{g}(\mathbf{x}_R) \tag{19}$$

for the state vector $\mathbf{x}_R = [u_B \;\; v_B \;\; w_B \;\; p_B \;\; q_B \;\; r_B \;\; \phi \;\; \theta \;\; \psi]^T$. As the dynamics of the heading angle is only affected by the dynamics of the other states, but not vice versa, its dynamics need not be considered for the computation of the reduced model. The input vector $\mathbf{c} = [\delta_E \; \delta_A \; \delta_R]^T$ is given by the elevator, the aileron and the rudder deflection angles. All nonlinear terms [17] like gravity terms, Euler terms, the nonlinear dependence between the differentiated Euler angles and the angular velocities and nonlinear aerodynamic terms are included in the vector:

$$\mathbf{g}(\mathbf{x}_R) = \begin{bmatrix} \sin\theta \\ \cos\theta\,\cos\phi \\ (q_B\,\sin\phi + r_B\,\cos\phi)\,\tan\theta \\ \frac{1}{\cos\theta}\,(q_B\,\sin\phi + r_B\,\cos\phi) \\ -r_B\,u_B + p_B\,w_B \\ q_B\,r_B \\ \vdots \end{bmatrix}. \tag{20}$$

The elastic motion is described relative to the rigid body motion. According to standard FEM and structural dynamic order reduction methods [7, 4, 9, 19] only a finite number of degrees of freedom, usually less than 100, have to be considered as a starting point for nonlinear model reduction:

$$\ddot{\mathbf{x}}_E + \mathbf{D}\dot{\mathbf{x}}_E + \mathbf{\Omega}\mathbf{x}_E = \mathbf{f}(\mathbf{x}_E, \dot{\mathbf{x}}_E, \mathbf{x}_R, \mathbf{c})\,. \tag{21}$$

$\mathbf{D}$ and $\mathbf{\Omega}$ are the structural damping and stiffness matrices. In case of generalized coordinates $\mathbf{\Omega}$ is diagonal. $\mathbf{f}(\mathbf{x}_E, \dot{\mathbf{x}}_E, \mathbf{x}_R, \mathbf{c})$ represents the aerodynamic forces. According to the aerodynamic design of the aircraft's geometry $\mathbf{f}(\mathbf{x}_E, \dot{\mathbf{x}}_E, \mathbf{x}_R, \mathbf{c})$ can be linearly approximated. Defining $\mathbf{v}_E = \dot{\mathbf{x}}_E$ yields a linear state space model for the elastic motion: $\dot{\mathbf{v}}_E = \mathbf{A}_{31}\mathbf{x}_R + \mathbf{A}_{32}\mathbf{x}_E + \mathbf{A}_{33}\mathbf{v}_E + \mathbf{B}_3\mathbf{c}$. The rigid body and the elastic motion are coupled via their aerodynamic interactions:

$$\begin{bmatrix} \dot{\mathbf{x}}_R \\ \dot{\mathbf{x}}_E \\ \dot{\mathbf{v}}_E \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} & \mathbf{A}_{13} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \\ \mathbf{A}_{31} & \mathbf{A}_{32} & \mathbf{A}_{33} \end{bmatrix} \begin{bmatrix} \mathbf{x}_R \\ \mathbf{x}_E \\ \mathbf{v}_E \end{bmatrix} + \begin{bmatrix} \mathbf{B}_1 \\ \mathbf{0} \\ \mathbf{B}_3 \end{bmatrix} \mathbf{c} + \begin{bmatrix} \mathbf{F} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \mathbf{g}(\mathbf{x}_R)\,, \tag{22}$$

where I is a unity matrix. The numerical matrix entries for the considered aircraft (type *B52E*) are given in [17, 2].

Consider damping of fuselage bending modes by active control [2, 17, 19, 12]. In case of a vanishing separation between the time scales of the elastic and the rigid body motion, an integral controller has to be designed, which incorporates standard flight mechanical control of the rigid body motion and mode control, [12, 19]. In [13] it was shown that already flight mechanical control of the rigid body motion of the *B52E* [17, 2] should consider a reduced order model of order 10 which contains 2 elastic states in addition to the 8 rigid body states. For integral flight mechanical and structural dynamic control of the *B52E* [17, 2] is here reduced to an order of $\tilde{n} = 16$. The goal is a model which adequately represents the dynamics of the lateral and the longitudinal load factors n_y, n_z and the flight mechanical quantities $\mathbf{x}_R$. The longitudinal and lateral load factor [17] include information about the rigid and the elastic aircraft motion and are therefore particularly suited for integral control of rigid and elastic motion. Via standard FEM and structural dynamic order reduction methods [7, 9, 17], the model can be condensed to an order of $n = 29$, retaining only the fuselage bending modes with the lowest frequencies [17, 2]. This model is the starting point for the nonlinear model reduction.

After having entered the system into the Toolbox, the nonlinear model reduction method has to be selected from the algorithm pulldown menu, figure 1. By a mouse click on the text field: "Load data set" (figure 1, left), one can load an already generated simulation data set of the original system for computing the reduced order system. By selecting "Simulate" one can interactively generate the data set, figure 1 (left): The main task of the engineer is to choose the input signals. All types of signals are supported, e.g. lookup tables or closed functions including standard input signals like steps, ramps, pulses, etc. The signal editor (figure 1, right) only requires to enter the simulation and signal parameters, which are then illustrated by the corresponding time histories. Following the listbox the user is prevented from entering an incomplete or inconsistent setup: Via mouse click one has to select all items from the top to the bottom of the listbox and then enter the required parameters in the input dialog window (figure 1, right). Figure 1, (right) shows, how to enter a multi doublet, [18].

Concerning the number and type of input signals in general one can take advantage of the analogy between model reduction and identification of dynamic systems: signals used for parameter identification also apply to the model reduction problem. To cover the whole operating domain of the system as densely as possible, the signals can be either chosen due to engineering judgement or in a more mathematical approach as a complete function system like sinoids, pulses, special polynomials (e.g. *Legendre* polynomials). As the systems under concern are nonlinear, both amplitude parameters and initial conditions have to be varied.

Here, several combinations of ramps and doublets, which are well–suited for identification of aerodynamic parameters [18] are used for all control surface deflections. Moreover, a combined turn– and $2.5g$–maneuver is considered, $g = 9.81\,m/s^2$. $2.5g$–maneuver means that a longitudinal load factor of $2.5g$ is generated. This combined maneuver excites most of the nonlinear terms, in particular the nonlinear coupling terms between longitudinal and lateral motion.

Next, the dominant states have to be chosen. Here, the nonlinear dominance analysis by *Lohmann* [15, 16] suggests a reduction to an order $\tilde{n} = 16$. The reduced order system of order $\tilde{n} = 16$ should be computed as a control-oriented realization: To have no additional nonlinear and no linear coupling terms between longitudinal and lateral motion in the reduced system, corresponding constraints $\mathbf{G}\,\tilde{\mathbf{E}}\,\mathbf{H} = \mathbf{0}$ are introduced. Thereby the model can be linearized at a set of trim points, yielding decoupled models for longitudinal and lateral motion. The idea is to design the longitudinal and the lateral controller independently and to design as many features as possible of these two controllers via linear methods, using the simple decoupled, linearized models of the reduced nonlinear system. Notice, that the nonlinear model reduction here ensures correct stationary performance of the linearized model by covering the stationary aeroelastic effects correctly. Afterwards, the combination of the two linear controllers has to be analyzed and appropriately extended based on the nonlinear reduced model.

In order to accelerate the reduction process, the assessment of a reduced order system for a large set of input signals and comparisons between the original model and different reduced order models can be done at different levels of precision via cost functionals. Cost functionals can be considered for (i) a whole set of N inputs, as in case of eq. (3), (ii) one simulation and (iii) the individual states of one

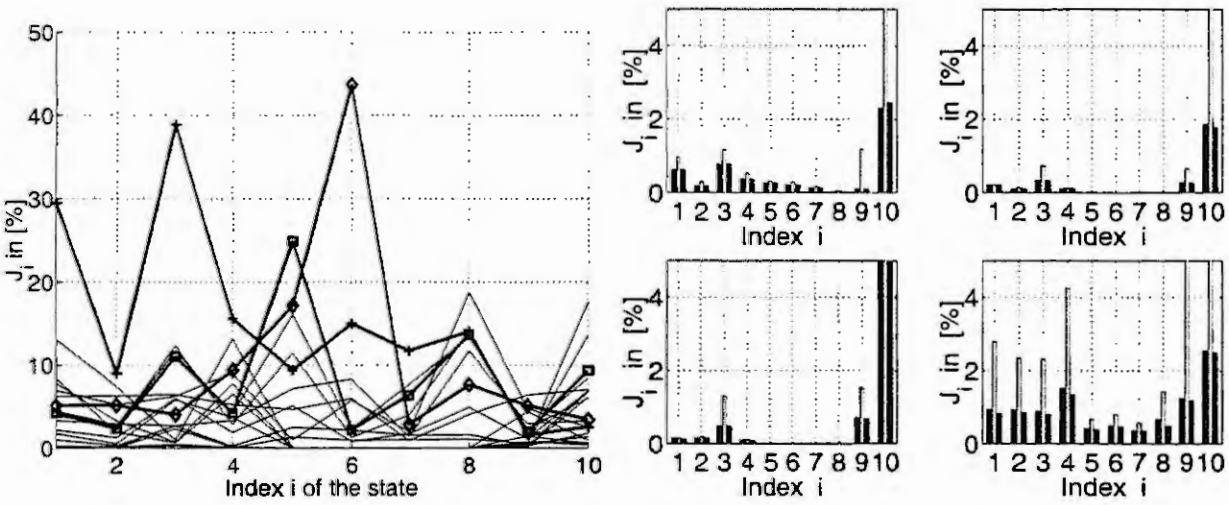


Figure 2: *Left:* Comparison of reduction results in parallel coordinates for an early iteration step of a cost functional based reduction for 20 test signals. Via polygons a large number of test signals can be considered, that were not used to compute the reduced system. Bad performing signals can easily be picked for the next iteration step of the model reduction process. The cost functionals measure the time-averaged deviations according to eq. (24) for the states i of the rigid body motion and of the first bending mode: 1=u_B, 2=w_B, 3=q_B, 4=θ, 5=v_B, 6=p_B, 7=r_B, 8=ϕ, 9=x_{E1}, 10=v_{E1}. Here, the three marked signals exceed a deviation of 20% for one or more states and are therefore used in the next iteration step. *Right:* Analysis of the reduced systems for four input signals after a suitable set of input signals has been found. Now, the above cost functionals J_i, eq. (24), are illustrated by error bars. Cost functional based reduction: left bar, *Hasenjäger* reduction: center bar, reduction by compensation: right bar. In the upper left figure the multi doublet of figure 1 (right) is considered as elevator input signal. This signal was also used for the computation of the reduced order system. In case of the other three subplots the input signals were not used for the cost functional based computation of the reduced order system.

simulation:

$$J = \frac{\int_0^{T_1+\ldots+T_N} ||\mathbf{x}_d(t) - \tilde{\mathbf{x}}(t)||^2 dt}{\int_0^{T_1+\ldots+T_N} ||\mathbf{x}_d(t)||^2 dt}, \quad J_\alpha = \frac{\int_0^{T_\alpha} ||\mathbf{x}_d(t) - \tilde{\mathbf{x}}(t)||^2 dt}{\int_0^{T_\alpha} ||\mathbf{x}_d(t)||^2 dt}, \tag{23}$$

$$J_j = \frac{\int_0^{T_\alpha} (x_{d,j}(t) - \tilde{x}_j(t))^2 dt}{\int_0^{T_\alpha} (x_{d,j}(t))^2 dt}. \tag{24}$$

The cost functionals are illustrated in parallel coordinates: The values of the cost functionals can be read from the y-axis. On the x-axis in case of (i) the index runs, which numbers the reduction method or the iteration number within one reduction process with a fixed method. In case of (ii) the index of the simulation and in case of (iii) the index of the state are given on the x-axis. In case of (ii) for comparison of methods or iteration steps and in case of (iii) for comparison of many test signals, several values occur at each index on the x-axis (figure 2).

Many test signals have to be analyzed in particular at the very beginning of model a reduction process and in the final assessment. To illustrate that conveniently, NEON offers both parallel bars (figure 2, right) and polygons (figure 2, left) for graphical comparison. Figure 2 (right) shows a comparative assessment of the reduced order system of order 16 for ten important states in terms of parallel bars for four test signals. Polygons (figure 2, left) are curves connecting the maximums of the bars from left to right via straight lines for the simulation of one test signal (figure 2, left) in case of (iii) and for one reduction method or one iteration step within the model *reduction process* in case of (ii). Now, each reduction method or each reduction step in case of (ii) and each test signal simulation (figure 2, left) in case of (iii) corresponds to one curve or polygon. Hereby, many simulations can be depicted in one diagram. In case of (ii), the worst and the best iteration step or reduction method for each test signal

and in case of (iii), the worst test signal for each state of the reduced order system can be easily grasped, by considering the envelope of all polygons.

Figure 2 (left) shows an assessment at the very beginning of the cost functional based reduction process. Concerning the deviations between the original and the reduced order system, three input signals constitute the upper envelope of all polygons. They are chosen for the computation of the reduced order system. They correspond to the fat polygons marked by symbols in figure 2 (left).

Next, the results of the model reduction of the structural dynamic aircraft model, having used both nonlinear reduction methods (section 2 and 3), are briefly presented in terms of parallel bars and time histories and compared with a *Hasenjäger* reduction. Concerning the *Hasenjäger* reduction, the linear part was reduced by the method of *Litz* [14] and the nonlinearities of the original system were transferred unchanged to the reduced order system. In figure 2, usually measured flight mechanical quantities and the longitudinal bending mode with the lowest frequency are depicted. These are the controlled quantities for integral flight mechanical and mode control. In the upper left of figure 2 (right), the multi doublet of figure 1, (right), was chosen as elevator input and was used for the computation of the reduced order system. The test signals of the other three subplots of figure 2 (right) were not considered for the

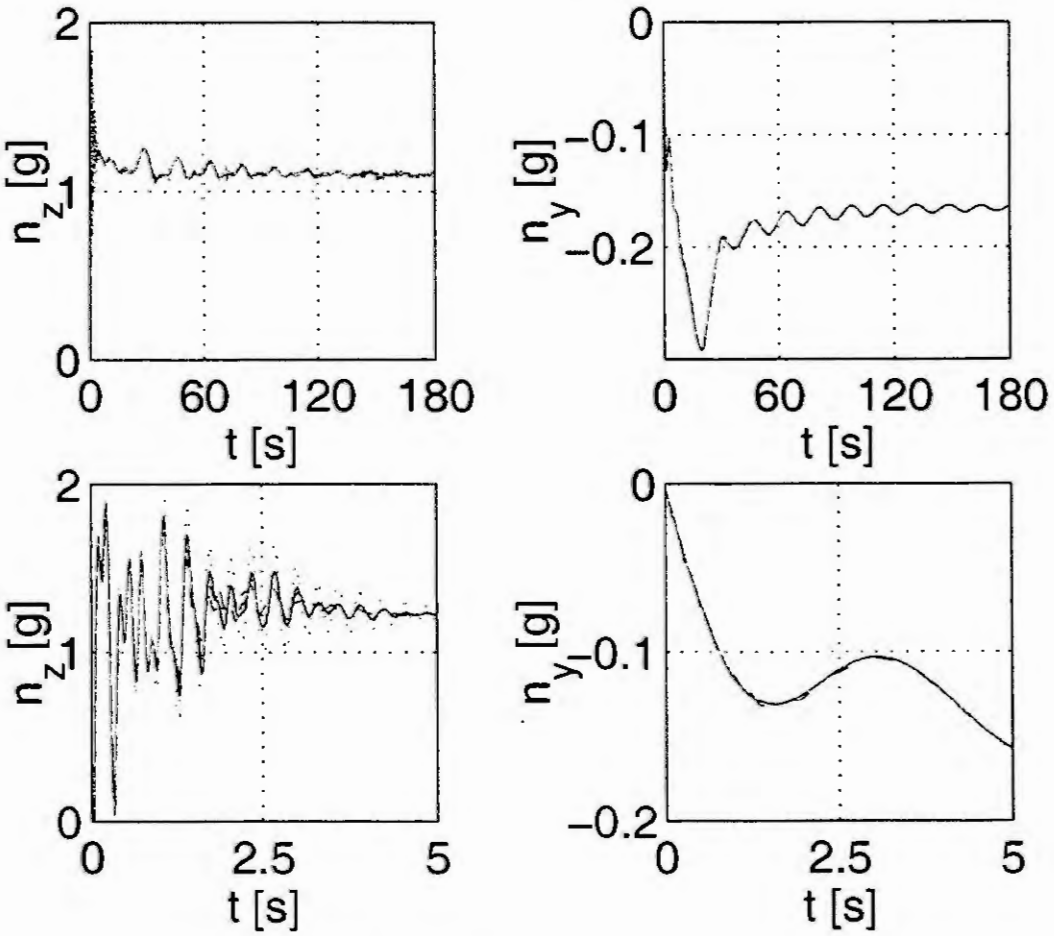


Figure 3: Comparison of longitudinal and lateral load factor n_z, n_y for a combined turn– and $2g$–maneuver: original system (–·), cost functional based reduction (–), reduction by compensation (- -), *Hasenjäger* reduction (··). Top row: complete time histories, bottom row: initial dynamics.

computation of the reduced order system. It turned out that the *cost functional based model reduction* and the *reduction by compensation* yield much better results than the *Hasenjäger* reduction (figure 2, right). This shows the necessity and the benefits of nonlinear model reduction methods. In figure 3, the time histories of the longitudinal and the lateral load factors n_y, n_z for a combined turn– and $2g$–maneuver are considered. This is an assessment maneuver to confirm that it was sufficient to consider

only the combined turn- and $2.5g$-maneuver for the computation of the reduced order system. Both nonlinear methods yield a better approximation of the transient amplitudes of n_z than the *Hasenjäger* reduction.

6. Conclusion:

The nonlinear model reduction methods which have here been developed are more consistent and adequate for nonlinear model reduction problems than linear and extended linear methods. In particular, they provide more design parameters in terms of the matrices F, W to approximate the original dynamics as well as possible. The Toolbox NEON has turned out to be suited to overcome burdens in the computer based application of the new methods. In case of a structural dynamic aircraft model it could be shown that the methods yield very low order nonlinear models suited for integral flight mechanical and structural dynamic controller design. The achieved accuracy is higher than that achieved by the *Hasenjäger* reduction.

References

[1] *Block, J. J.; Stragnac T. W.*: Applied Active Control for a Nonlinear Aeroelastic Structure. Journal of Guidance, Control and Dynamics, Vol. 21, No. 6, 1998, pp. 838-845.

[2] *Burris, P. M.; Bender, M. A.*: Aircraft Load Alleviation and Mode Stabilization (LAMS) — B52, system analysis, synthesis and design. AFFDL-TR-68-161. WPAFB, Dayton, Ohio, 1968.

[3] *Fife, P.*: Transition layers in singular perturbation problems. J. Diff. Equations, 1974, p. 77-105.

[4] *Försching, H. W.*: Foundations of Aeroelasticity. Springer–Verlag, New York, 1974.

[5] *Freemann, R. A.; Kokotović, P. V.*: Robust Nonlinear Controller Design. Birkenhäuser–Verlag, Boston, 1996, pp. 101-136.

[6] *Hasenjäger, E.*: Digitale Zustandsregelung von Parabolantennen unter Berücksichtigung von Nichtlinearitäten. VDI–Verlag, Düsseldorf, 1991.

[7] *He, J.; Ewins, D. J.*: Compatibility of Measured and Predicted Vibration Modes in Model Improvement Studies. AIAA Journal, Vol.29, No.5, September–October, 1991, pp. 798–803.

[8] *Isidori, A.*: Nonlinear Control Systems. 3d ed., Springer–Verlag, London, 1995.

[9] *Karpel, M.*: Reduced–Order Aeroelastic Models via Dynamic Residualization. AIAA Journal of Aircraft, Vol. 27, May, 1990, pp. 449-455.

[10] *Kokotović, P. V.; Khalil, H. K.*: Singular Perturbations in Systems and Control. IEEE Press, New York, 1986.

[11] *Kokotović, P. V.*: Constructive Nonlinear Control: Progress in the 90's. Proceedings of 14th World Congress of IFAC, Beijing, P.R.China, July, 1999.

[12] *König, K.; Schuler, J.*: Integral Control of Large Flexible Aircraft. RTO (AGARD) Symposium, Ottawa, Canada, 1999.

[13] *Kordt, M.*: Nonlinear Order Reduction of a High Capacity Aircraft. Automatisierungstechnik 47, 1999.

[14] *Litz, L.*: Reduktion der Ordnung linearer Zustandsraummodelle mittels modaler Verfahren. Hochschulverlag, Freiburg 1979.

[15] *Lohmann, B.*: Order Reduction and Dominance Analysis of Nonlinear Systems. Automatisierungstechnik 42, 1994, pp. 466-474.

[16] *Lohmann, B.*: Order Reduction and Dominance Analysis of Nonlinear Systems. Proceedings of 2nd MATHMOD, Vienna, 1997.

[17] *McLean, D.*: Automatic Flight Control Systems. Prentice Hall International (U.K.) Ltd., London, 1990, pp. 102-126, 423-9.

[18] *Mehra, R. K.*: Optimal Inputs for Linear System Identification. IEEE Transactions on Automatic Control, Vol. AC–19, No. 3, June 1974.

[19] *Schuler, J.*: Flugregelung und aktive Schwingungsdaempfung für flexible Grossraumflugzeuge — Modellbildung und Simulation. Fortschrittberichte VDI, Reihe 8: Mess-, Steuerungs- und Regelungstechnik, Nr. 688. Düsseldorf, VDI Verlag, 1998.

[20] *Sepulchre, R.; Janković, M.; Kokotović, P. V.*: Constructive Nonlinear Control. Springer–Verlag, New York, 1997, pp. 276-283.

POLYTOPIC LINEAR MODELING OF A CLASS OF NONLINEAR SYSTEMS: AN AUTOMATIC MODEL GENERATING METHOD

G.Z. Angelis[1], M.J.G. van de Molengraft[1], J. Verstraete[2], J.J. Kok[1]
[1]Systems and Control Group, Eindhoven University of Technology, The Netherlands
g.z.angelis@tue.nl
[2]A.S.M.Lithography, Veldhoven, The Netherlands

Abstract. Polytopic linear models (PLMs) are models with parameters that vary within a polytope of the model parameter space. These models are also known as (Takagi-Sugeno) Fuzzy models, Local Model Networks or Multimodels. The PLM model class is rich and has already shown to be useful for controller synthesis. Therefore it is of great importance to develop methods that, given a nonlinear system model and/or observed data, construct a PLM suitable for controller synthesis.

We have developed a novel method that, given a sufficient smooth nonlinear continuous-time state space description of the system, automatically generates a PLM that is close to this system. The modeling method is implemented in Matlab and case studies indicate the practical applicability of the method. The method automatically constructs the simplest PLM with the desired accuracy.

Introduction

The objective of this paper is to approximate the system:

$$\Sigma : \begin{cases} \dot{x} &= f(x,u) \\ y &= h(x,u) \end{cases}$$

with state $x \in X \subseteq \mathbb{R}^n$, input $u \in U \subseteq \mathbb{R}^m$ and output $y \in Y \subseteq \mathbb{R}^p$, by a PLM description:

$$\text{PLM} : \begin{cases} \dot{x} = \sum_{i=1}^{N_m} w_i(z)\{A_i x + B_i u + a_i\} \\ y = \sum_{i=1}^{N_m} w_i(z)\{C_i x + D_i u + c_i\} \end{cases}, \quad \sum_{i=1}^{N_m} \omega_i(z) = 1,\ \omega_i(z) \geq 0$$

The model is called a PLM since the model consists of a set of parametrizations of non-homogenous linear models, i.e. $M_i = \left(\begin{array}{c|c|c} A_i & B_i & a_i \\ \hline C_i & D_i & c_i \end{array}\right)$ with $i = \{1, ..., N_m\}$, that define a polytope in the model parameter-space. Conceptually, the PLM can be thought of as based on N_m linearizations of Σ in an operating point $\psi \in \Psi = X \times U$. If the parametrization M_i is a locally valid description in an operating region $\Psi_i \subset \Psi$ then it is likely that the PLM will give a qualitative good description of Σ globally, i.e. within the region Ψ, if $\Psi \subseteq (\cup \Psi_i)$. Here $z = h(\psi)$, is a set of variables that schedule the locally valid models M_i in such a way that $w_i(z) \simeq 1$ if $z \in h(\Psi_i)$, and $w_i(z) = 0$ elsewhere. PLMs occur frequently in literature and although they all have an equivalent mathematical structure they are given different names i.e. Fuzzy Models [6], Multi-Models [4] or Local Model Networks [3].

The model structure has several desirable attributes that can be exploited. Firstly, the model class is rich since a large class of nonlinear systems can be approximated arbitrarily close (this will be formalized later) with the proposed model structure as proved in [3]. As a result a large class of nonlinear systems can be parametrized as PLMs. Secondly, since the model is based on 'multiple linearizations' that define 'multiple operating regions', the system can be qualitatively interpreted in terms of these operating regimes. Finally, well founded synthesis results are reported, and these are emerging rapidly for PLM systems [2][5][1].

In the current situation it is of great importance to develop methods that, given a nonlinear system model and/or observed data, construct a PLM sufficiently close to the real system, that is suitable for controller synthesis.

We have developed a novel method that, given Σ automatically generates a PLM such that

$$d_{\Sigma PLM} := \max_{\psi \in \Psi} \|\Sigma_{rhs}(\psi) - \text{PLM}_{rhs}(\psi)\|_2 \leq \varepsilon$$

with $\varepsilon > 0$, here L_{rhs} means right-hand-side of model L and $\|\cdot\|_2$ denotes the Euclidean norm. The particular choice of distance between systems enables the following proposition to be made:

Proposition 1 *Given $f \in C^1(E)$, E an open subset of $\mathbb{R}^{n+m}$ and $\Psi = \{\psi \mid |\psi_i - d_i| \leq e_i, i = 1, \cdots, n+m\}$, d_i, e_i positive constants, and such that $\Psi \subset E$, then there exists a rhs of a PLM with*

$$N_m = \Pi_{i=1}^{n+m} \left\lceil \frac{L_1 \sqrt{n}}{\varepsilon} e_i \right\rceil \tag{1}$$

finite such that $d_{\Sigma PLM} \leq \varepsilon$, $\varepsilon > 0$ arbitrary. The ceiling operator $\lceil . \rceil : \mathbb{R} \to \mathbb{N}_+$ maps a real number to the smallest integer greater than or equal to that number. Here L_1 is such that $\|f(\psi) - f_i(\psi)\|_2 \leq L_1 \|\psi - \psi_{0i}\|_2$ with $f_i(\psi) = f(\psi_{0i}) + \frac{\partial f}{\partial \psi}(\psi_{0i})(\psi - \psi_{0i})$. Furthermore, in that case the solutions $\xi(t)$ of Σ and $\zeta(t)$ of the PLM originating from $\xi(0) = \zeta(0)$ on the interval $[0, t]$ are uniformly close in the following sense: $\|\xi(t) - \zeta(t)\|_2 \leq \varepsilon \frac{e^{L_2 t} - 1}{L_2}$, when L_2 is an upperbound on $\left\| \frac{\partial f}{\partial \psi} \right\|_2$.

The modeling problem can be divided into three subproblems, which are solved sequentially. This is done in the next three sections. A more detailed description of the modeling method can be found in [7].

From operating space to scheduling space

When the operating space is of high dimension, the curse of dimensionality will restrict the applicability of the PLMing approach. The core of the problem is that the number of operating regimes, needed to uniformly partition the operating space, increases exponentially with the dimension of the operating space, see (1). However, in some cases the models can be scheduled on a space of lower dimension, which will reduce the modeling problem considerably. In these cases the structure of the system is exploited to reduce dimensionality [3]. It will be shown that an exponential reduction of the number of models N_m involved in the PLM description can be achieved. The idea is to determine a set of variables $z = h(\psi)$ that projects ψ onto a lower dimensional space Z, the so called scheduling space. This is formulated in the following proposition:

Proposition 2 *Given the state equation of Σ with $f(\psi_L, \psi_N) = f_1(\psi_N) + f_2(\psi_N)\psi_L$ where $f \in \mathbb{R}^n$ and $\psi_L \in \Psi_L$, $\psi_N \in \Psi_N$, $\Psi_L \times \Psi_N = \Psi$. Assume $f \in C^1(E)$, E an open subset of $\mathbb{R}^{n+m}$ and any compact set $\Psi \subset E$. Then it is necessary to take at least $z = \psi_N$, and it suffices to take $PLM_{rhs}(\psi) = \sum_{i=1}^{N_m} w_i(z) g_i(\psi_L)$ to achieve that $d_{\Sigma PLM} \leq \varepsilon$, $\varepsilon > 0$ arbitrary, and with N_m finite. Furthermore, if $f_2(\psi_N) = F$ and $f_1(\psi_N) \in C^2(E_1)$, E_1 an open subset of $\mathbb{R}^{n_Z}$ and f_1 has $n - n_N$ scalar linear components, then within any set Ψ, with $\Psi_N = \{\psi_N \mid |\psi_{Ni} - d_i| \leq e_i, i = 1, \cdots, n_Z\}$ compact such that $\Psi_N \subset E_1$ there exists a $PLM_{rhs}(\psi) = \sum_{i=1}^{N_m} w_i(z) g_i(\psi)$ with $N_m = \Pi_{i=1}^{n_z} \left\lceil \frac{e_i}{\sqrt{2\varepsilon}} \sqrt{|\lambda_\xi| n_N^{1/2} n_z} \right\rceil$ that achieves ε- accuracy. Here $|\lambda_\xi|$ is an upperbound on the second order Taylor remainder, a measure for the nonlinearity of the system.*

Proposition 2 covers a large class of systems. For instance, for many mechanical systems $\psi_N = x$ the state, and $\psi_L = u$ the input. The proposition is based on a worst case scenario, since it is assumed that the maximum nonlinearity, as measured with the Taylor remainder, can occur everywhere in the scheduling space. The number of models N_m depends on the smallest required scheduling regime Z_i, because the scheduling space is uniformly partitioned. Furthermore, from the proof it follows that the locally valid models can be chosen as Taylor linearizations and the scheduling functions can be chosen as basis functions with compact support.

In the next step the number of models is reduced by dropping the idea of uniform partitioning. The key assumption is that nonlinearity, as measured with a norm bound the Taylor remainder (linearization error), depends on the scheduling variable z. In the process of reduction the computation of N_m will play an important role.

From scheduling space to scheduling regimes

Two procedures will be introduced; segregation and aggregation, that automatically decompose Z in qualitatively different scheduling regimes Z_i such that $Z \subseteq (\cup_i Z_i)$

The aggregation procedure starts from a uniformly decomposed scheduling space, as defined in the previous section and depicted in Figure 1a, and then tries to unite scheduling regimes conceptually as

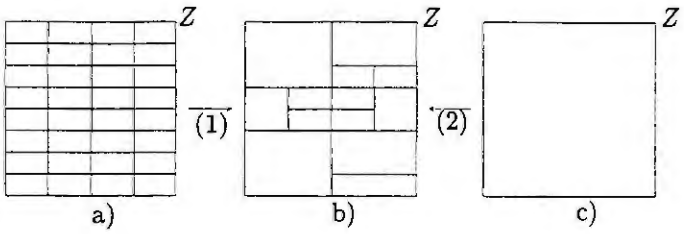


Figure 1: The procedures aggregation (1) and segregation (2) for a two dimensional scheduling space Z.

follows: if $N_m(Z_i \cup Z_j) = 1$ for two adjoining scheduling regimes Z_i and Z_j then unite these regimes into a new scheduling regime. The model is simplified since N_m is reduced. This action is repeated until no further model simplification occurs. Contrary to the aggregation procedure, the segregation procedure starts from one simple linear model which covers the entire scheduling space as depicted in Figure 1c, and step by step model complexity is increased conceptually as follows: split the n_Z dimensional scheduling space 'in the middle' in two scheduling regimes Z_i and Z_{i+n_Z}. This can be done in n_Z different ways, i.e. $i = \{1, ..n_Z\}$. Compute the number of models $N_m = N_m(Z_i) + N_m(Z_{i+n_Z})$ for all obtained configurations that is sufficient to achieve ε-accuracy. Select the configuration with the lowest number of models. One segregation step is then completed. Segregation is repeated until $N_m(Z_j) = 1$.

From scheduling regimes to PLM parameters

For every regime Z_i a linear model M_i is parametrized together with the scheduling function $w_i(z)$. The scheduling functions are chosen as normalized basis functions, centered in the middle of operating regime i and in accordance with the size of scheduling regime Z_i. The parameters of the linear models are obtained by linearization of the system in the centers of the scheduling space.

Example

The method has some interesting features. Firstly, it significantly reduces the total number of models needed to achieve ε-accuracy. Secondly, the operating regions are chosen such that they reflect qualitatively different behavior. The modeling method was automated, as a Matlab implementation, and evaluated on several case studies. The following example, PLMing a rotating arm with friction, will illustrate the method and its features.

Consider a mechanical system consisting of a rotating arm subjected to significant friction in the driveline:

$$\Sigma : \begin{cases} \dot{x}_1 = f_1(\psi) = x_2 \\ \dot{x}_2 = f_2(\psi) = -\frac{T_c}{J}\frac{2}{\pi}\arctan(\kappa x_2) - \frac{T_s - T_c}{J} e^{-\left(\frac{x_2}{v_s}\right)^2} \frac{2}{\pi}\arctan(\kappa x_2) - \frac{\sigma_2}{J} x_2 + \frac{c_m}{J} u \end{cases}$$

with inertia $J = 0.0292$ $[Nm\,\mathrm{sec}^2]$, motor constant $c_m = 16$ [], and friction parameters $T_c = 0.416$ $[Nm]$, $T_s = 0.4657$ $[Nm]$, $v_s = 0.2$ $[rad/\,\mathrm{sec}]$, $\sigma_2 = 0.0135$ $[Nm\,\mathrm{sec}]$, $\kappa = 1000$[].

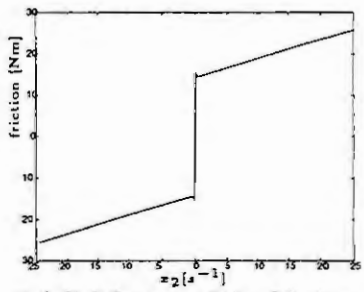
We demand $d_{\Sigma PLM} \leq 2$ and assume $|x_2| \leq 25[rad/\,\mathrm{sec}]$. Proposition 2 is applied with $n_Z = 1$, $|\lambda_\xi| = 54$ and $n_N = 1$. We obtain 92 models uniformly covering and dividing the scheduling space. Since λ_ξ varies as a function of x_2 we can use the procedure aggregatión or segregation to reduce the number of models. For both the aggregation or segregation procedure the number of models is reduced to 6. Here, the result of the segregation method was improved by an extra aggregation step. The advantage of segregation over aggregation, is the smaller computational effort. The obtained center $d(Z_i)$, the width $2e(Z_i)$ and modelparameters of each scheduling regime Z_i : $d(Z_i) - e(Z_i) \leq x_2 \leq d(Z_i) + e(Z_i)$ are summarized in Table 1. The triples $\left(A_i = \begin{pmatrix} 0 & 1 \\ 0 & \hat{A}_i \end{pmatrix}, B_i = \begin{pmatrix} 0 \\ \hat{B}_i \end{pmatrix}, a_i = \begin{pmatrix} 0 \\ \hat{a}_i \end{pmatrix}\right)$ are determined by linearizing $f(\psi)$ in the point μ_{0i}. Here, $\mu_{0i} = [\ 0 \quad d(Z_i) \quad 0\ ]^T$. As we could expect $\hat{B}_i$ is the same for the six operating regimes, because Σ_{rhs} is linear on u. The scheduling functions are chosen as normalized radial basis functions that are placed in the center $d(Z_i)$ of each operating regime Z_i,

Table 1: Operating regimes and model parameters

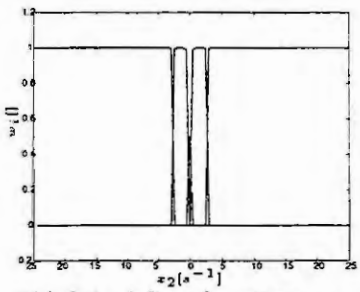
	Z_1	Z_2	Z_3	Z_4	Z_5	Z_6
$d(Z_i)$	−13.8889	−1.6204	−0.2315	0.2315	1.6204	13.8889
$e(Z_i)$	11.1111	1.1574	0.2315	0.2315	1.1574	11.1111
$\hat{A}_i$	−0.4624	−0.4658	4.5093	4.5093	−0.4658	−0.4624
$\hat{B}_i$	547.9452	547.9452	547.9452	547.9452	547.9452	547.9452
$\hat{a}_i$	14.2453	14.2354	15.8029	−15.8029	−14.2354	−14.2453

i.e. $w_i(x_2) = \frac{\rho_i(x_2)}{\sum_{i=1}^{N_m} \rho_j(x_2)}$ with $\rho_i(x_2) = e^{-\frac{(x_2 - d(Z_i))^2}{2\gamma e(Z_i)}}$. The user-specified parameter γ is chosen 0.25, indicating almost no overlap between the models.

The friction torque of the PLM is validated and depicted together with the corresponding scheduling functions in Figure 2. A visually identical friction curve compared to Σ is observed. Regime $Z_1$and Z_6 indicate the viscous friction. Regimes Z_3 and Z_4 indicate the coulomb friction.



2a) Validation of the friction



2b) Scheduling functions w_i

Summary and Conclusions

A method is presented that constructs a PLM, with desired accuracy, given a sufficient smooth non-linear continuous time state-space system. The simplest PLM is selected. A case study indicates the practical applicability of the method. In the near future we will focus on extensions of the method such that a PLM can be derived from experimental data of the real system.

References

[1] G.Z. Angelis, M.J.G. van de Molengraft, R.J.P. van der Linden, and J.J. Kok. Robust controller design and performance for polytopic models. In *Proceedings of the ECC*, September 1999.

[2] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*. SIAM, Philadelphia, 1994. ISBN 0-89871-334-X.

[3] T.A. Johansen. *Operating Regime based Process Modeling and Identification*. PhD thesis, Department of Engineering Cybernetics, The norwegian Institute of Technology, University of Trondheim, 1994.

[4] R. Murray-Smith and T.A. Johansen, editors. *Multiple Model Approaches to Modelling and Control*. Taylor and Francis, London, 1997.

[5] O. Slupphaug and B.A. Foss. Robust stabilization of discrete-time multi-model systems using piecewise affine state-feedback. In *DYCOPS-5, Corfu, Greece*, pages 47–52. IFAC, 1998.

[6] M. Sugeno and G.T. Kang. Structure identification of fuzzy model. *Fuzzy Sets and Systems*, 26:15–33, 1988.

[7] J. Verstraete. Polytopic linear modeling of nonlinear mechanical systems. Systems and Control Group, Eindhoven university of Technology, The Netherlands, Master Thesis, 1999.

EFFICIENT NUMERICAL MODEL REDUCTION METHODS FOR DISCRETE-TIME SYSTEMS

Peter Benner[1], Enrique S. Quintana-Ortí[2], and Gregorio Quintana-Ortí[2]
[1]Zentrum für Technomathematik, Fachbereich 3
Universität Bremen, D–28334 Bremen (Germany)
[2]Departamento de Informática
Universidad Jaime I, 12080 Castellón (Spain)

Abstract. Model reduction is of fundamental importance in many modeling and control applications. Here we address algorithmic aspects of model reduction methods based on balanced truncation of linear discrete-time systems. The methods require the computation of the Gramians of the system. Using an accelerated fixed point iteration for computing the full-rank factors of the Gramians yields some favorable computational aspects, particularly for non-minimal systems. The computations only require efficient implementations of basic linear algebra operations readily available on modern computer architectures.

Introduction

Consider the transfer function matrix (TFM) $G(\lambda) = C(\lambda I - A)^{-1}B + D$, and the associated stable, but not necessarily minimal, realization of a discrete, linear time-invariant (LTI) system,

$$x_{k+1} = Ax_k + Bu_k, \quad y_k = Cx_k + Du_k, \qquad k = 0, 1, 2, \ldots, \tag{1}$$

where $x_0 = \hat{x}$ is given and $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$, $D \in \mathbb{R}^{p\times m}$. The number of state variables n is said to be the order of the system. We assume that the spectrum of A as denoted by $\Lambda(A)$ is contained in the open unit disk, i.e., A is *(Schur) stable* or *convergent*. This implies that all the poles of the TFM $G(s)$ are contained in the open unit disk and hence the stability of the system (1).

We are interested in finding a reduced order LTI system,

$$\tilde{x}_{k+1} = \tilde{A}\tilde{x}_k + \tilde{B}\tilde{u}_k, \quad \tilde{y}_k = \tilde{C}\tilde{x}_k + \tilde{D}\tilde{u}_k, \qquad k = 0, 1, 2, \ldots, \tag{2}$$

of order ℓ, $\ell \ll n$, with $\tilde{x}_0 = \hat{\tilde{x}}$, such that $\tilde{G}(\lambda) = \tilde{C}(\lambda I - \tilde{A})^{-1}\tilde{B} + \tilde{D}$ approximates $G(\lambda)$.

There is no general technique for model reduction that can be considered as optimal in an overall sense since the reliability, performance and adequacy of the reduced system strongly depends on the system characteristics. The methods considered here are all based on balanced truncation (BT) methods [5, 8].

BT model reduction methods are based on information retrieved from the controllability and observability Gramians W_c and W_o, respectively, of the system (1). These are given by the solutions of two "coupled" (as they share the same coefficient matrix A) *Stein equations* (or *discrete Lyapunov equations*)

$$AW_cA^T - W_c + BB^T = 0, \qquad A^TW_oA - W_o + C^TC = 0. \tag{3}$$

As A is assumed to be stable, W_c and W_o are positive semidefinite and therefore can be factored as $W_c = S^TS$ and $W_o = R^TR$. The factors S and R are called the *Cholesky factors* of the Gramians. These factors are usually computed using Hammarling's method [3], yielding $S, R \in \mathbb{R}^{n\times n}$ upper triangular. Numerically reliable model reduction methods use these Cholesky factors rather than the Gramians themselves; see, e.g., [8, 9, 10]. However, if the system is not minimal, then W_c and/or W_o are singular and the Cholesky factors computed by this method are not full-rank matrices. In particular, for large systems, it can often be observed that the *numerical rank* of the Cholesky factors is much less than n.

We therefore describe in the next section a method based on an accelerated fixed point iteration that compute *full-rank factorizations* $W_c = \hat{S}^T\hat{S}$ and $\hat{W}_o = \hat{R}^T\hat{R}$, i.e., $\hat{S} \in \mathbb{R}^{\mathrm{rank}(W_c)\times n}$, $\hat{R} \in \mathbb{R}^{\mathrm{rank}(W_o)\times n}$. This can save a significant amount of computational cost and workspace solving the Stein equations and particularly in the subsequent computations for computing the reduced-order model. This approach also has some advantages regarding numerical robustness when determining the McMillan degree and a minimal realization of an LTI system.

We then describe the numerical implementation of model reduction methods based on balanced truncation using the full-rank factors of the Gramians. Numerical examples reporting the accuracy and performance of the resulting routines on serial and parallel computers will be provided in an extended version of this note[1]. Using parallel computers with distributed memory such as Linux-PC or workstation clusters allows the application of our methods to systems up to order $n = \mathcal{O}(10^5)$.

[1]Available from `http://www.math.uni-bremen.de/zetem/berichte.html`.

Computing the Gramians

Consider the Stein equations in (3). Both can be formulated in a fixed point form $X = FXF^T + G$, from which it is straightforward to derive a fixed point iteration. This iteration converges to X if $\rho(F) < 1$, where $\rho(F)$ denotes the spectral radius of F. That is, convergence is guaranteed under the given assumptions. The convergence rate of this iteration is linear. A quadratically convergent version of the fixed point iteration is suggested in [7]. Setting $X_0 := G$, $F_0 := F$, this iteration can be written as

$$X_{k+1} := F_k X_k F_k^T + X_k, \quad F_{k+1} := F_k^2, \qquad k = 0, 1, 2, \ldots. \tag{4}$$

The above iteration is referred to as the *squared Smith iteration*. As the two equations in (3) share the same coefficient matrix A, we can derive a coupled iteration to solve both equations simultaneously.

$$\begin{array}{rlrlrll} X_0 & := BB^T, & Y_0 & := C^TC, & A_0 & := A, & \\ X_{k+1} & := A_k X_k A_k^T + X_k, & Y_{k+1} & := A_k^T Y_k A_k + Y_k, & A_{k+1} & := A_k^2, & k = 0, 1, 2, \ldots \end{array} \tag{5}$$

The most appealing feature of the squared Smith iteration regarding its implementation is that all the computational cost comes from matrix products. These can be implemented very efficiently on modern serial and parallel computers.

The convergence theory of the Smith iteration (4) derived in [7] yields that for $\rho(F) < 1$ there exist real constants $0 < \mu$ and $0 < \rho < 1$ such that $\|X - X_k\|_2 \leq \mu \|G\|_2 (1-\rho)^{-1} \rho^{2^k}$. This shows that the method converges for all equations with Schur stable coefficient matrices F.

In the case considered here, the "right-hand sides" of the Stein equations are positive semidefinite and are given in factored form BB^T and C^TC. As A is stable, Lyapunov stability theory (see, e.g., [4]) shows that the solution matrices are positive semidefinite and hence can be factored as $W_c = S^TS$, $W_o = R^TR$. The factors $S \in \mathbb{R}^{s \times n}$ and $R \in \mathbb{R}^{r \times n}$ are called the Cholesky factor of the solution. Usually, $s = r = n$ such that S, R are square, possibly singular, matrices [3]. Here we will also use "Cholesky factor" to denote a *full-rank factor* of the solution, i.e., $\operatorname{rank}(S) = \operatorname{rank}(W_c) = s \leq n$, $\operatorname{rank}(R) = \operatorname{rank}(W_o) = r \leq n$. This has several advantages. The condition number of the solution matrix can be up to the square of that of its Cholesky factor. Hence, a significant increase in accuracy can often be observed working with the factor if the solution matrix is ill-conditioned. Moreover, for the case $r, s \ll n$ we will show in the next section that significant savings in computational work are obtained by using the full-rank factors rather than the square Cholesky factors for subsequent computations.

The coupled squared Smith iteration (5) can be modified to compute the full-rank factors of W_c, W_o directly; see [1]. We focus here on the iteration for computing W_c; the iteration for W_o can be treated analogously. Setting $G = BB^T$, the X_k iteration in (5) can be re-written by setting $S_0 := B$ and

$$S_{k+1}S_{k+1}^T \leftarrow S_kS_k^T + A_k(S_kS_k^T)A_k^T = [\,S_k,\ A_kS_k\,] \begin{bmatrix} S_k^T \\ S_k^TA_k^T \end{bmatrix}, \quad \text{for } k = 0, 1, 2, \ldots. \tag{6}$$

In each step (6) the current iterate S_k is augmented by A_kS_k such that $S_{k+1} := [S_k,\ S_kA_k]$.

The above approach requires to double in each iteration step the workspace needed for the iterates L_k. Two approaches are possible to limit the required workspace to a fixed size [1]. We will focus here on one approach which is particularly appealing for the purpose of model reduction.

In each iteration step, we can compute a rank-revealing LQ factorization (see, e.g., [2, Chapter 5]) $\tilde{S}_{k+1} := \frac{1}{\sqrt{2}}[\,S_k,\ S_kA_k\,] = Q\hat{S}_{k+1}\Pi_{k+1}^T$. In that case, the next iterate $S_{k+1} \in \mathbb{R}^{n \times \operatorname{rank}(\tilde{S}_{k+1})}$ is obtained as the left $n \times \operatorname{rank}(\tilde{S}_{k+1})$ part of the product of the permutation matrix Π_{k+1} and the lower triangular matrix $\hat{S}_{k+1}$, i.e.,

$$\begin{bmatrix} (\Pi_{k+1})_{11} & (\Pi_{k+1})_{12} \\ (\Pi_{k+1})_{21} & (\Pi_{k+1})_{22} \end{bmatrix} \begin{bmatrix} (\hat{S}_{k+1})_{11} & 0 \\ (\hat{S}_{k+1})_{21} & 0 \end{bmatrix} =: [\,S_{k+1},\ 0\,],$$

starting from S_0 obtained by a rank-revealing LQ factorization of B. It follows that $\tilde{S}_{k+1}\tilde{S}_{k+1}^T = S_{k+1}S_{k+1}^T$ and $\sqrt{2}S = (\lim_{k\to\infty} S_k)^T$.

As $\operatorname{rank}(W_c)$ may be up to n, this requires a work space of size up to $2n \times n$. On the other hand, the (numerical) rank of the full-rank factors is often much less than n. Hence, the computational cost of performing LQ factorizations is well balanced by keeping the number of columns of S_{k+1} small. Usually, the computational cost of this approach is less than the cost of Hammarling's method [3], see [1] for details. This approach can also be used to compute low-rank approximations to the full-rank factor by either increasing the tolerance threshold for determining the numerical rank or by fixing the allowed number of columns in S_k.

Balanced Truncation Model Reduction using Full-Rank Factors

In [8] it is shown that BT model reduction can be achieved using SR^T instead of the product of the Gramians themselves. Here, S and R denote the square, possibly singular Cholesky factors of W_c and W_o, respectively. The resulting *square-root (SR) method* avoids working with the Gramians as their condition number can be up to the square of the condition number of the Cholesky factors. The first step in the SR method is to compute the SVD

$$SR^T = [U_1\ U_2] \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}. \tag{7}$$

Here, the matrices are partitioned at a given dimension ℓ with $\Sigma_1 = \operatorname{diag}(\sigma_1, \ldots, \sigma_\ell)$ and $\Sigma_2 = \operatorname{diag}(\sigma_{\ell+1}, \ldots, \sigma_n)$ such that $\sigma_1 \geq \sigma_2 \geq \ldots \sigma_\ell > \sigma_{\ell+1} \geq \sigma_{r+2} \geq \ldots \geq \sigma_n \geq 0$. If $\sigma_\ell > 0$ and $\sigma_{\ell+1} = 0$, i.e., $\Sigma_2 = 0$, then r is the *McMillan* degree of the given discrete-time LTI system. That is, r is the state-space dimension of a minimal realization of the system.

For model reduction, r should be chosen in order to give a natural separation of the states, i.e., one should look in the Hankel singular values σ_k, $k = 1, \ldots, n$, for a large gap $\sigma_\ell \gg \sigma_{\ell+1}$ [8].

So far we have assumed that the Cholesky factors S and R of the Gramians are square $n \times n$ matrices. For non-minimal systems, we have $\operatorname{rank}(S) < n$ and/or $\operatorname{rank}(R) < n$. Hence, rather than working with the Cholesky factors, we may use the full-rank factors $\hat{S}$, $\hat{R}$ of W_c, W_o that are computed when using the method described in the last section. The SVD in (7) can then be obtained from that of $\hat{S}\hat{R}^T$ as follows. Here we assume $\operatorname{rank}(\hat{S}) =: s \geq r := \operatorname{rank}(\hat{R})$, the case $s < r$ can be treated analogously. Then we can compute the SVD

$$\hat{S}\hat{R}^T = \hat{U} \begin{bmatrix} \hat{\Sigma} \\ 0 \end{bmatrix} \hat{V}^T, \qquad \hat{\Sigma} = \operatorname{diag}(\sigma_1, \ldots, \sigma_r), \tag{8}$$

where $\hat{U} \in \mathbb{R}^{s \times s}$, $\hat{V} \in \mathbb{R}^{r \times r}$. Now let $\ell \leq r$ be the order of the reduced (or minimal) system. Partitioning $\hat{U} = \left[\begin{array}{cc} \hat{U}_1 & \hat{U}_2 \end{array}\right]$ such that $\hat{U}_1 \in \mathbb{R}^{s \times \ell}$, $\hat{U}_2 \in \mathbb{R}^{s \times s-\ell}$, $\hat{V}_1 \in \mathbb{R}^{r \times \ell}$, $\hat{V}_2 \in \mathbb{R}^{r \times r-\ell}$, $\hat{\Sigma}_1 = \operatorname{diag}(\sigma_1, \ldots, \sigma_\ell)$, and $\hat{\Sigma}_2 = \operatorname{diag}(\sigma_{\ell+1}, \ldots, \sigma_r)$, the SVD of SR^T is given by

$$SR^T = \left[\begin{array}{c|cc} \hat{U}_1 & \hat{U}_2 & 0 \\ \hline 0 & 0 & I_{n-s} \end{array}\right] \left[\begin{array}{c|cc} \hat{\Sigma}_1 & 0 & 0 \\ \hline 0 & \hat{\Sigma}_2 & 0 \\ 0 & 0 & 0 \end{array}\right] \left[\begin{array}{c|cc} \hat{V}_1 & \hat{V}_2 & 0 \\ \hline 0 & 0 & I_{n-s} \end{array}\right]^T. \tag{9}$$

We will see that all the subsequent computations can also be performed just working with $\hat{U}_1$, $\hat{\Sigma}_1$, and $\hat{V}_1$ rather than using the data from the full-size SVD in (9). This amounts in a significant savings of workspace and computational cost. For example, using the Golub–Reinsch SVD (see, e.g., [2]), (7) requires $22n^3$ flops and workspace for $2n^2$ real numbers if U, V are to be formed explicitly while (8) only requires $14sr^2 + 8r^3$ flops and workspace for $s^2 + r^2$ real numbers. In particular, for large-scale dynamical systems, the *numerical rank* of W_c, W_o and $\hat{S}$, $\hat{R}$ is often much less than n. Suppose that (numerically) $s = r = n/10$, then the computation of (8) is 1000 times less expensive than that of (7) and only 1% of the workspace is required for (8) as compared to (7).

Defining

$$T_l = \Sigma_1^{-1/2} V_1^T R = \hat{\Sigma}_1^{-1/2} \hat{V}_1^T \hat{R} \qquad \text{and} \qquad T_r = S^T U_1 \Sigma_1^{-1/2} = \hat{S}^T \hat{U}_1 \hat{\Sigma}_1^{-1/2}, \tag{10}$$

the reduced system (2) is given by

$$\tilde{A} = T_l A T_r, \quad \tilde{B} = T_l B, \quad \tilde{C} = C T_r, \quad \text{and} \quad \tilde{D} = D. \tag{11}$$

In case that $\Sigma_1 > 0$ and $\Sigma_2 = 0$, (11) is a *minimal realization* of the TFM $G(\lambda)$ [8], i.e., ℓ is the minimum dimension of the state-space for which a realization of $G(\lambda)$ in the form of an LTI system (1) is possible. Hence, choosing ℓ in (7) maximal such that $\sigma_\ell > 0$ and $\sigma_{\ell+1} = 0$, this procedure can be used to compute minimal realizations if the decision "$\sigma_{\ell+1} = 0$" is based on a numerically reliable criterion.

It can further be proved that for a stable LTI system, choosing any partitioning in (7) such that $\sigma_\ell > \sigma_{\ell+1}$ yields a stable, minimal, and balanced reduced model. The Gramians corresponding to the resulting TFM $\tilde{G}(\lambda)$ are both equal to Σ_1. See [8] for a proof.

As the reduced model in (11) is balanced, the projection matrices in (10) tend to be ill-conditioned if the original system is highly unbalanced, resulting in inaccurate reduced-order models. An alternative here are *balancing-free (BF) methods* [6] for which the reduced-order model is not balanced. The *balancing-free square-root (BFSR) method* combines the best characteristics of the SR and BF ap-

proaches [9, 10]. It shares the first two steps (solving the equations in (3) for the Cholesky factors and computing the SVD in (7)) with the SR method described above. Then, two "skinny" QR factorizations are computed,

$$S^T U_1 = \hat{S}^T \hat{U}_1 = P T_S, \qquad R^T V_1 = \hat{R}^T \hat{V}_1 = Q T_R,$$

where P, $Q \in \mathbb{R}^{n \times \ell}$ have orthonormal columns and T_S, $T_R \in \mathbb{R}^{\ell \times \ell}$ are upper triangular. The reduced system is then given as in (11) with the projection matrices defined by $T_l = (Q^T P)^{-1} Q^T$ and $T_r = P$.

We have implemented only the SR and BFSR algorithms as the BF algorithm described in [6] usually shows no advantage over BFSR algorithms with respect to model reduction abilities. Moreover, the BF approach is potentially numerically unstable. For one, it uses the product $W_c W_o$ rather than SR^T, leading to a squaring of the condition number of the matrix product. Second, the projection matrices T_l and T_r computed by the BFSR approach are often significantly better conditioned than those computed by the BF approach [9, 10]. Furthermore, both SR and BFSR algorithms can be efficiently parallelized while the BF method needs a parallelized version of the QR algorithm with re-ordering of eigenvalues. This presents severe implementation difficulties; see, e.g., [1] for a discussion of this topic. Implementation details as well as accuracy and performance details are available in an extended version of this note[2].

Concluding Remarks

We have described efficient and reliable numerical algorithms for the realization of model reduction methods based on the square-root version of balanced truncation. Using the full-rank factors of the Gramians often enhances the efficiency and accuracy of these methods significantly. Implementations of the discussed methods are based on highly optimized software packages for numerical linear algebra on serial and parallel computers. Parallel computing allows to use these methods for systems of state-space dimension up to order $\mathcal{O}(10^5)$.

References

[1] P. Benner, E.S. Quintana-Ortí, and G. Quintana-Ortí. Numerical solution of Schur stable linear matrix equations on multicomputers. Berichte aus der Technomathematik, Report 99-13[2], FB3 - Mathematik und Informatik, Universität Bremen, D-28334 Bremen, November 1999.

[2] G.H. Golub and C.F. Van Loan. *Matrix Computations.* Johns Hopkins University Press, Baltimore, third edition, 1996.

[3] S.J. Hammarling. Numerical solution of the discrete-time, convergent, non-negative definite Lyapunov equation. *Sys. Control Lett.*, 17 (1991), 137–139.

[4] P. Lancaster and M. Tismenetsky. *The Theory of Matrices.* Academic Press, Orlando, 2nd edition, 1985.

[5] B.C. Moore. Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE Trans. Automat. Control*, AC-26 (1981), 17–32.

[6] M.G. Safonov and R.Y. Chiang. A Schur method for balanced-truncation model reduction. *IEEE Trans. Automat. Control*, 34 (1989), 729–733.

[7] R.A. Smith. Matrix equation $XA + BX = C$. *SIAM J. Appl. Math.*, 16 (1968), 198–201.

[8] M.S. Tombs and I. Postlethwaite. Truncated balanced realization of a stable non-minimal state-space system. *Internat. J. Control*, 46 (1987), 1319–1330.

[9] A. Varga. Efficient minimal realization procedure based on balancing. In: Prepr. IMACS Symp. on Modelling and Control of Technological Systems, 1991, vol. 2, 42–47.

[10] A. Varga. Model reduction routines for SLICOT. NICONET Report 1999-8[3], The Working Group on Software (WGS), June 1999.

[2] Available from `http://www.math.uni-bremen.de/zetem/berichte.html`.

[3] Available from `http://www.win.tue.nl/niconet/NIC2/reports.html`.

REDUCED ORDER FEEDBACK DESIGN FOR HIGH INDEX SINGULARLY PERTURBED SYSTEMS.

S.A. Mikhailov and P.C. Müller
Safety Control Egineering
University of Wuppertal, Gauß str. 20, D-42097 Wuppertal, Germany
E-mails: mihailov@uni-wuppertal.de, mueller@srm.uni-wuppertal.de

Abstract We consider singular singularly perturbed systems (SSPS) where the reduced order differential-algebraic system has index > 1. Using terminology of descriptor systems literature, we will call these systems high index singularly perturbed systems. Tikhonov-Levinson theory and standard time-scale modeling do not apply for the SSPS. Therefore the related problems have to be considered in the course of linear-quadratic optimal control design. These difficulties can be alleviated by means of decoupling the slow and fast motions in SSPS. We analyze the general structure of optimal linear-quadratic control design.

Introduction

In the last decades there was an essential progress dealing with index 1 singularly perturbed dynamical systems. But still there are many unsolved problems related to analysis and design of high index singularly perturbed systems. In these problems we run into various types of singularities. The first one is connected with small parameters in the coefficients of the derivatives in the differential equations. The second singularity shows up if the reduced DAE system has index > 1. And the last singularity is due to singularity of the weighting matrix in the performance criterion. The behavior of high index systems is quite different from index 1 singularly perturbed systems: the reduced system has another order, the boundary layer system is of higher order depending on system index, and for the controlled systems a new characteristic of properness has to be taken into consideration. Proper and non-proper systems distinguish between the cases if the system is exclusively governed by the control input or by its higher-order time-derivatives additionally. In this paper, introductory results for optimal reduced order control design of linear high index singularly perturbed systems with respect to an infinite-horizon quadratic performance criterion will be presented.

Consider linear singular perturbed system

$$\dot{x}_1 = A_{11}x_1 + A_{12}x_2 + B_1u \tag{1}$$

$$\varepsilon\dot{x}_2 = A_{21}x_1 + A_{22}x_2 + B_2u \tag{2}$$

where ε is a small positive scalar, $x_1 \in R^n$ is the "slow" state vector, $x_2 \in R^m$ is the "fast" state vector, $u \in R^k$ is the vector of control variables. The essential feature of high index systems is that A_{22} is a singular matrix.

For the control design the quadratic performance criterion

$$J = \tfrac{1}{2}\int_0^\infty \begin{bmatrix} x_1 \\ x_2 \\ u \end{bmatrix}^T \begin{bmatrix} Z_{11} & Z_{12} & Z_{13} \\ Z_{12}^T & Z_{22} & Z_{23} \\ Z_{12}^T & Z_{23}^T & Z_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ u \end{bmatrix} dt \Rightarrow \min_u \tag{3}$$

is considered where

$$Z_{33} > 0, \qquad \begin{bmatrix} Z_{11} & Z_{12} & Z_{13} \\ Z_{12}^T & Z_{22} & Z_{23} \\ Z_{12}^T & Z_{23}^T & Z_{33} \end{bmatrix} \geq 0 \tag{4}$$

is assumed. The problem to be solved consists in designing a (feedback) control u which minimizes (3) having regard to the dynamic system (1)-(2).

Index

The index measures the type of singularity. By analogy with linear descriptor systems in semi-explicit form the notion of (uniform) index k for the linear singular perturbed systems can be introduced as follows [4]:

$$\begin{aligned} k=1 \quad &: \quad A_{22} \ regular \\ k=2 \quad &: \quad A_{22}=0, \quad A_{21}A_{12} \ regular \\ k\geq 3 \quad &: \quad A_{22}=0, \quad A_{21}A_{11}^{j}A_{12}=0, \quad j=0,\cdots,k-3, \quad A_{21}A_{11}^{k-3}A_{12} \ regular \end{aligned} \tag{5}$$

The index gives the number of times the algebraic equations of a reduced system have to be differentiated to get a full set of differential equations for fast variables. "Hidden" constraints generated by process differentiation appear for the systems with index > 1.

Properness

In the following we distinguish the two cases where the solution depend either only on $u(t)$ but not on its derivatives $\dot{u},\cdots,u^{(k-1)}$ or on $u(t)$ and its derivatives $\dot{u},\cdots,u^{(k-1)}$. In the first case the system is called "proper", in the second case "non-proper". In case of (5) the system (1)-(2) is proper if and only if [4]

$$\begin{aligned} k=1 \quad &: \quad always \\ k=2 \quad &: \quad B_2=0 \\ k\geq 3 \quad &: \quad B_2=0, \quad A_{21}A_{11}^{j}B_1=0, \quad j=0,\cdots,k-3. \end{aligned} \tag{6}$$

Properness is significant for the control design. For "proper" singularly perturbed systems optimization can be applied regularly. For "non-proper" systems situation is quite different and extension of state variables must be carried out to deal correctly with the influence of time-derivatives of the control input.

The decoupling of the slow and fast variables (index 2 problem)

For index 1 system by means of Riccati transformation [2] it is possible to obtain the uncoupled equations for fast and slow variables. For pure index 2 problem $A_{22}=0$ and $\det(A_{21}A_{12})\neq 0$ hold, and Riccati transformation is not applicable. Introducing the complementary rank $(n-m)$ projection [1]

$$P=I-Q \tag{7}$$

where $$Q=A_{12}(A_{21}A_{12})^{-1}A_{21}, \quad rank(Q)=m<n, \quad PA_{12}=A_{21}P=PQ=0,$$

then we look for the n - vector x_1 as the sum

$$x_1=Px_1+Qx_1=v+w. \tag{8}$$

After transformations of co-ordinate (8) we obtain the following reduced order system ($\varepsilon=0$)

$$\dot{v}=PA_{11}v+P\{B_1-A_{11}(A_{21}A_{12})^{-1}A_{12}B_2\}u \tag{9}$$

$$w=-(A_{21}A_{12})^{-1}A_{12}B_2u \tag{10}$$

$$x_2=-(A_{21}A_{12})^{-1}(A_{21}A_{11}x_1+A_{21}B_1u+B_2\dot{u}) \tag{11}$$

Differential-algebraic system (9)-(11) is the reduced order system for the original equations(1)-(2). It consists of the slow subsystem of order $n-m$, and $2m$ algebraic equations. The decomposition into slow and algebraic subsystems suggests that separate slow and boundary control laws are designed for each subsystem, and then combined into a composite design for the original system. Note, that for index 1 problem the slow subsystem is of order n, the boundary layer subsystem consists of m algebraic equations [2].

Solution of the control problem

In this section we discuss general structure of linear-quadratic optimal synthesis. Although the SSPS description (1), (2) shows explicitly only input u, there are hidden effects related to time derivatives $\dot{u}$ as it shown by (11).

This situation is very different from the common state-space discussions. It is necessary to distinguish between the two cases.

Proper systems. For the proper systems a standard linear-quadratic control problem for the reduced system can be constructed. The control for the slow subsystem (9) has to be designed with respect to the criterion (3). In the case of proper systems it is possible to substitute fast variables w and x_2 from (10), (11) into functional (3). After replacement we arrive in the common linear-quadratic problem. Therefore the solution is obtained by the "Riccati" procedure for the system of reduced order $n-m$. The optimal control is a proportional feedback of the states of the slow subsystem (9).

Non-proper systems. For non-proper systems the Riccati equation approach does not meet the problem directly. An extension of state and control variables has to be carried out:

$$\xi = u, \quad w_e = \dot{u}. \tag{12}$$

Here, w_e is considered as a new control input vector. Introducing an extended state vector

$$x_e = \begin{bmatrix} v \\ \xi \end{bmatrix}, \tag{13}$$

an extended dynamical system can be described including the dynamics of slow subsystem (9) and of the extension (12):

$$\dot{x}_e = A_e x_e + B_e w_e \tag{14}$$

Substituting x_2 form (11) in the performance criterion (3) and taking into account (12), (13), a linear-quadratic optimal problem with respect to x_e and w_e appears. But obviously a singular control problem is obtained. There is not a regular weighting of the new input w_e. One way is to construct the direct solution of this singular control problem. But this solution has rather theoretical meaning due to the impulses in $\dot{u}(t)$. The second way is to regularize the problem introducing an additional weighting matrix in the performance criterion. In this case we have a standard linear-quadratic problem for the extended system (14) of order $n-m+1$, cf. the analogue problems for the descriptor systems [3].

Conclusions

In this paper an approach to analyze and design linear time invariant singular singularly perturbed system has been presented. It was necessary to introduce the notions of proper and non-proper systems and to distinguish between two cases. For the proper systems the usual Riccati approach can be applied to the slow subsystem resulting in a proportional state feedback of the variables of the slow subsystem. For non-proper SSPS extended state variables and a new control input have to be introduced. Additionally, the performance criterion has to be regularized to obtain a regular optimal control problem. Then a Riccati problem of an extended system has to be solved. The result is a dynamic feedback with respect to the slow subsystem.

References

1. Kalachev, L.V., O'Malley, R.,E. The regularization of linear differential-algebraic equations. SIAM J: Math. Anal., (1996), Vol. 27, No. 1, pp. 258-273.
2. Mikhailov, S.A. Müller, P.C. Near-time-optimal feedback control of mechanical systems with fast and slow motions. in: D.H. van Campen (ed.): IUTAM Symposium on Interaction Between Dynamics and Control in Advanced Mechanical Systems, Kluwer Academic Publishers, Dordrecht 1997, pp. 239-246.
3. Müller, P.C., Linear-Quadratic Optimal Regulator for Descriptor Systems. In Proc. 15th IMACS Wold Congress on Scientific Computation, Modelling and Applied Mathematics (Ed. Sydow, A.), Vol. 5, Berlin, 1997, pp. 49-54.
4. Müller, P.C. Stability and optimal control of nonlinear descriptor systems: a survey. Appl. Math. and Com. Sci., (1998), vol. 8, no. 2, pp. 269-286.

Minimal complexity approximating models of multiport systems

Patrick Dewilde
DIMES - Delft University of Technology
Delft, The Netherlands.

Abstract

The paper treats bounded linear operators in a computational context. These operators act between spaces of sequences of vectors with ℓ_2 norms in such a way that the sequence of treatment is strictly respected. Under these assumptions we treat the question on how a given operator T can be approximated by one for which the operational complexity is minimal given a certain tolerance. It turns out that the classical Hankel norm model reduction theory generalizes. The result is now dependent on the singular values not of a single Hankel matrix but of an intertwined collection of them. Generalized interpolation theory in the Schur-Takagi fashion inducing J-unitary operators play a central role in the development of the theory, a complete account of which can be found in the recently published book [2].

1 Introduction and definitions

The problem we wish to consider is the relation between approximation and complexity. The context is 'linear operators' en we put ourselves in a computational situation, where the notion of 'complexity' plays a central role. How can we tackle the connexion?

An operator maps an 'input' to an 'output' - $y = uT$. u and y will be sequences of data, vectors or more generally sequences of vectors. T is a general operator which respects the order of data, i.e. the output data at position k is only dependent on the input data before k. We say: T is a causal map. We interpret the order requirement in a 'strong' sense: once the input data before k is known, y_k must be computed. In this way we rule out the possibility of 'divide and conquer' methods which lead to minimal complexity computations in the sense of absolute total number of computations.

Our formalism is as follows: the sequence of input spaces is called $\mathcal{M}_k$, each of which is finite dimensional. Likewise, the sequence of output spaces is $\mathcal{N}_k$. Dimensions may be zero, in which case they are reduces to 'placeholders'. The product of a matrix of dimensions $m \times 0$ with one of dimensions $0 \times n$ is an $m \times n$ matrix of zeros. We allow T to be an infinite operator, but assume it bounded in the ℓ_2 sense, mapping $\ell_2^{\mathcal{M}}$ to $\ell_2^{\mathcal{N}}$. With the causality assumption in place, T has a matrix representation:

$$T = \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \iddots \\ 0 & T_{-1,0} & T_{-1,1} & T_{-1,2} & \cdots \\ & \boxed{0} & T_{0,1} & T_{0,2} & \cdots \\ & & 0 & T_{1,2} & \cdots \\ & & & \ddots & \end{bmatrix},$$

but more general operators will occur as well.

Since our main theme is 'approximation', besides T we need an accuracy measure. To cater for generality, instead of introducing an 'epsilon' we allow for an hermitian diagonal matrix, which we call Γ. The central problem that we address is then: given T and Γ, find an operator T_a of low numerical complexity such that, for some strong norm,

$$\|\Gamma^{-1}(T - T_a)\| < 1.$$

The question of 'which strong norm to use' is also important, our strategy will be to consider the norm and the complexity question together. Suppose that T is a finite matrix, and that the computation starts

at $k = 0$, then we would have in sequence:

$$\begin{array}{rcl} y_1 & = & u_0 T_{0,1} \\ y_2 & = & u_0 T_{0,2} + u_1 T_{1,2} \\ y_3 & = & u_0 T_{0,3} + u_1 T_{1,3} + u_2 T_{2,3} \\ & etc & \end{array}$$

The complexity of this way of computing increases linearly with k, we would need an always growing memory when k progresses, and the number of computations would be like kn^2 where n is a cap on the size of the matrices $T_{k,i}$. It seems that at stage k we have to take $u_0, \cdots, u_{k-1}$ in memory, while also the size of this vector determines the number of operations. What is to be taken in memory to be used at stage k we call the 'state vector' x_k, and the number of computations is of course a direct function of its dimension. An approach (that we shall motivate further at the end of the paper) is to take the minimal dimension needed for x_k as complexity measure at stage k.

2 Time-varying systems

What we need is a generalisation of the classical Kronecker theory. If we position ourselves at a point k of the sequence, then the minimal state complexity needed for the computation at that point is given by the rank of the past-to-future map at that time point. This map is

$$H_k = \begin{bmatrix} \vdots & \vdots & \iddots \\ T_{k-2,k} & T_{k-2,k+1} & \cdots \\ T_{k-1,k} & T_{k-1,k+1} & \cdots \end{bmatrix}$$

We call it the *Hankel map* at point k. If each Hankel map is finite dimensional, then the system can be realised using finite computations based on a state at stage k of dimension equal to its rank - such a system we call *locally finite*. A minimal realisation is obtained via an otherwise arbitrary factorisation of the Hankel map:

$$H_k = \mathcal{R}_k \mathcal{O}_k$$

and will have the form

$$\begin{array}{rcl} x_{k+1} & = & x_k A_k + u_k B_k \\ y_k & = & x_k C_k + u_k D_k. \end{array}$$

In fact, we have the (defining) recursions

$$\mathcal{O}_k = [C_k \;\; A_k \mathcal{O}_{k+1}], \; \mathcal{R}_k = \begin{bmatrix} A_{k-1}\mathcal{R}_{k-1} \\ B_{k-1} \end{bmatrix}.$$

It is convenient to view H_k as part of a global Hankel operator linking an extended input space to an extended output space. Let $\mathcal{X}_2^{\mathcal{M}}$ consist of a stack of sequences, for each k one, and endowed with a Hilbert-Schmidt metric. For $\mathcal{X}_2^{\mathcal{N}}$ similarly. T can be viewed as a map $\mathcal{X}_2^{\mathcal{M}} \to \mathcal{X}_2^{\mathcal{N}} : \;\; Y_{k,\cdot} = U_{k,\cdot} T$. Let any $\mathcal{X}_2$ be further orthogonally decomposed as $\mathcal{X}_2 = \mathcal{U}_2 \oplus \mathcal{U}_2'$ in which $\mathcal{U}_2$ consists of uppers (i.e. $U_{k,i} = 0$ when $k < i$) and $\mathcal{U}_2'$ of strictly lowers. Let furthermore $\mathbf{P}$ indicate the projection on uppers, and $\mathbf{P}'$ on lowers, then the global Hankel operator is given by $\cdot H_T = \cdot \mathbf{P}' T \mathbf{P}$.

For a strictly upper operator such as T we can define its Hankel norm as $\|T\|_H \triangleq \|H_T\| = \sup_k \|H_k\|$. This is a strong norm, but not as strong as the operator norm $\|T\|$.

For convenient further elaboration, we need some shorthand notation as follows. The realisation matrices can be assembled in diagonal (or 'instantaneous') operators: $A = \mathrm{diag}(\cdots \;\; A_k \;\; \cdots)$ etc... Let Z be the 'causal shift', then a finitely realisable operator T can be represented by $T = D + BZ(I - AZ)^{-1}C$ provided proper meaning can be given to the inverse. We say that the realisation is 'uniformily exponentially stable' if the spectral radius ℓ_A of AZ is less than one. In that case, $(I - AZ)^{-1}$ is a bounded operator. A realisation is not unique. If $\{A_k, B_k, C_k, D_k\}$ is one, then another one is given by $\{R_k^{-1} A_k R_{k+1}, B_k R_{k+1}, R_k^{-1} C_k, D_k\}$, in which R_k is a non-singular state transformation ($x_k = x_k' R_k$). If, in addition, the R_k^{-1} are uniformily bounded, then the transformation is called a Lyapunov transformation. The ues property is stable under Lyapunov transformations with the same spectral radius.

3 External factorization

We say that a system is *inner* if it is upper and unitary. We say that T has a left external factorization if there exist an inner operator U_ℓ and an upper operator Δ_ℓ such that

$$T = \Delta_\ell^* U_\ell.$$

In general, T will not have an external factorization. If T has a ues state space realisation, then a left external factorisation will exist if the recursive Lyapunov equations

$$\Lambda_k = A_k \Lambda_{k+1} A_k^* + C_k^* C_k$$

have a solution which is strictly positive definite ($\exists \epsilon > 0 : \quad \forall k(\Lambda_k > \epsilon)$), in which case one may choose $\Lambda_k = R_k R_k^*$ as state transformation and the resulting realisation $\{A_k', B_k', C_k', D_k\}$ will be in so called *output normal form*, i.e. $A_k' A_k'^* + C_k C_k'^* = I$. Realisations for U_ℓ is then found as

$$(U_\ell)_k = \begin{bmatrix} A_k' & C_k' \\ B_{U,k} & D_{U,k} \end{bmatrix},$$

in which $[B_{U,k} \quad D_{U,k}]$ form a unitary completion of $[A_k' \quad C_k']$ for each k. A ues system for which Λ_k is strictly positive is said to be *strictly observable*, and Λ_k is called its observability Gramian.

4 The approximation procedure

We shall attempt to find a minimal approximating T_a for T in Hankel norm (given Γ). The basic procedure is as follows:
Step 1: external factorization: find a minimal left external factorization for T: $T = \Delta_\ell^* U_\ell$.
Step 2: interpolation step: let $\mathcal{M}_U$ be the input space of U and define the signature matrix

$$J \triangleq \begin{bmatrix} I_{\mathcal{M}_U} & \\ & -I_{\mathcal{M}} \end{bmatrix}.$$

Determine a bounded and minimal J-unitary, upper operator Θ such that

$$[U^* \quad -T^*\Gamma^{-1}]\Theta = [A' \quad -B'] \tag{1}$$

is upper.
Step 3: Nehari approximant: define $T' = \Gamma\Theta_{22}^{-*}B'^* = T - \Gamma(\Theta_{12}\Theta_{22}^{-1})^* U$.
Step 4: Hankel norm approximant: let T_a be the upper part of T'.
Then $\|\Gamma^{-1}(T - T_a)\|_H < I$ and T_a has minimal state complexity.

To see that these steps indeed solve the problem, more explanation is needed, besides showing their feasibility. We give those in order:
1. T does not have in general a left coprime factorization. However, that is not needed. A bounded T can be approximated as closely as one wishes with a ues system that is strictly observable, for example by using an adequate series expansion in Z.
2. The interpolation step is in fact also a special kind of external factorisation. When one analyses it, one obtains a non-singularity condition on an appropriate Gramian for its solution. The Gramian involved is:

$$M_{k+1} = A_k^* M_k A_k + B_k^* \Gamma_k^{-2} B_k$$

and the determination of Θ will be possible if each matrix $I - M_k$ is non-singular. If that is the case, let $\tilde{J}_k$ be its inertia and X_k such that

$$X_k^* \tilde{J} X_k = I - M_k$$

then a J-unitary and ues realization for Θ_k is a J-unitary completion of

$$\begin{bmatrix} X_k A_k X_{k+1}^{-1} \\ B_{U,k} X_{k+1}^{-1} \\ \Gamma_k^{-1} B_k X_{k+1}^{-1} \end{bmatrix}$$

(which is easily determined numerically). It follows that the solution will exist if there is a kind of dichotomy on the singular values of the Hankel matrix $(H_{\Gamma^{-1}T})_k$: partition its singular values according to whether they are smaller than one or larger: $(\sigma_+)_{i,k} < 1$, and say N_k such that $(\sigma_-)_{i,k} > 1$. If such a partition away from 1 exists uniformily over k, then Θ exists.
3. From the J-unitarity of Θ we find that Θ_{22} is invertible, and that $\|\Theta_{12}\Theta_{22}^{-1}\| < 1$. Working out (1) and using the definition $T' = \Gamma\Theta_{22}^{-*}B'^{*}$ we find

$$T - T' = \Gamma(\Theta_{12}\Theta_{22}^{-1})^{*}U.$$

It follows immediately that $\|T - T'\| < 1$. However, T' is not an upper operator. There is a 'time-varying version of Nehari's theorem' which states that if T_a = strict upper part of T', then $\|T - T_a\|_H = \|T - T'\|$. Hence T_a is an adequate approximant of T.
4. Next is the question of the complexity of the approximant. It turns out that N_k is the local degree of Θ_{22}^{-*} and also of $\Theta_{22}^{-*}B'^{*}$. This degree is also minimal. We have the following theorem:

Theorem 1 *Let T be strictly upper, ues, finitely realisable and strictly observable. Suppose that the singular values of the $(H_{\Gamma^{-1}T})_k$ are clustered away from 1 so that $\sup_{i,k}(\sigma_+)_{i,k} < 1$ and $\inf_{i,k}(\sigma_-)_{i,k} > 1$. Suppose moreover that the number of $(\sigma_+)_{i,k}$ is N_k. Then there exists a strictly upper T_a such that $\|T - T_a\|_H < 1$ of local degree N_k. There is no approximant of lower local degree.*

5 Discussion

In case the original set up is finite dimensional, then the approximation problem stated is always solvable. It may be necessary to make a slight adaptation on Γ to avoid a non-singular Θ. The singular case is solvable but rather complex. It has been worked out in [1]. The more general case where the system is infinite but first time-invariant, then becomes time-varying and then moves on to become time-invariant again (the IVI case) is solvable in the same sense. In the more general case, more powerful mathematical tools may be needed but they go beyond the computational scope envisaged here.

All in all, it turns out that in computational cases one can indeed find approximations in Hankel norm with minimal state complexity given the tolerance Γ. Remains the question whether minimal state complexity indeed corresponds to low or even minimal computational complexity. Although we do not know the complete solution to this problem, we are able to solve an important intermediate step. It turns out that, given maximal time-varying freedom of coefficients in the original operator (i.e. all coefficients in T are free), then there exist minimal realisations that are also algebraically minimal, i.e. that utilise the minimum number of free parameters.

References

[1] P. Dewilde. J-unitary matrices for algebraic approximation and interpolation - the singular case. In M. Moonen and B. De Moor, editors, *SVD and Signal Processing, III, Algorithms, Architectures and Applications*, pages 209–223. Elsevier, 1995.

[2] P. Dewilde and A.-J. van der Veen. *Time-varying Systems and Computations.* Kluwer, 1998.

AN EXTENSION OF SCATTERING VARIABLES TO SPATIAL MECHANISMS

B.M.J. Maschke [1] [2], A.J. van der Schaft [1] and C. Bidard [3]
[1] Systems, Signals and Control Department, Faculty of Mathematical Sciences,
University of Twente, P.O.Box 217, 7500 AE ENSCHEDE, The Netherlands.
e-mail:maschke, A.J.vanderSchaft@math.utwente.nl
[2] Laboratoire d'Automatisme Industriel, Conservatoire National des Arts et Métiers
21 rue Pinel, 75013 PARIS, France.
[3] Department of Signals and Systems, Faculty of Electrical Engineering
and Institute for Biomedical Technology BMTI,
University of Twente, P.O.Box 217, 7500 AE ENSCHEDE, The Netherlands.

Abstract. In this paper we firstly recall a definition of scattering variables which uses no inner product but solely the duality product between dual vector spaces. Secondly we show how this definition allows to derive a definition of scattering variables for kinestatic models of mechanisms which is invariant under the adjoint representation.

1. Introduction

Scattering variables are a classical tool in electrical circuit synthesis [2] and in control of passive systems [13]. They arise in particular for the control of telemanipulator where a transmission line model is integrated in the model in order to represent communication delays between the master and the slave robot [1]. There, the mechanical models considered there are one–dimensional, hence the velocity and force variables are scalars and the definition of the scattering variables matches exactly with the definition used for electrical circuits which is based on some, arbitrary defined, inner product [2]. In the case of spatial mechanisms, the velocity and force variables become twists and wrenches [16]. However, it is known that, for these variables, there exist no inner product invariant with respect to rigid body displacement (i.e. with respect to the adjoint representation map) [16]. Hence the definition of the scattering variables for spatial mechanisms cannot be extended from the definition for one–dimensional mechanical systems.

In this paper we shall firstly recall an alternative definition of scattering variables proposed in [12] [13] in relation with the definition of Dirac structures [5]. Dirac structures are the geometry of the state space of constrained or implicit Hamiltonian systems which generalizes the Poisson bracket and the pre–symplectic forms [5]. But they arise also in so–called implicit port controlled Hamiltonian systems [14] associated with network models of engineering systems where they represent the admissible set for the interconnection variables (called power variables) of a power continuous interconnection networks [4] [10] [11] [14] [15]. Secondly we shall use the definition of scattering variables recalled in the first part, in order to derive a definition of scattering variables for spatial mechanisms using solely the duality product. As a consequence we shall show that this definition is invariant with respect to the adjoint representation (i.e. to rigid body displacements). We shall briefly give a scattering representation of two constitutive parts of kinestatic models: the port connection graph of a mechanisms and a kinematic pair.

2. Dirac structures and scattering variables on vector spaces

2. 1. Dirac structures on vector spaces

Let $\mathcal{V}$ denote a real vector space of dimension n and $\mathcal{V}^*$ its dual vector space (i.e. the space of 1–forms over $\mathcal{V}$). The scattering variables of $\mathcal{V}$ are defined as elements of subspaces of the direct sum $\mathcal{V} \oplus \mathcal{V}^*$ with respect to the following *canonical symmetric tensor* (called + *pairing* by Courant [5]).

Definition 1 : Canonical symmetric tensor

For any two pairs (v_1, w_1) *and* (v_2, w_2) *in* $\mathcal{V} \oplus \mathcal{V}^*$, *define the symmetric tensor:*

$$<(v_1, w_1), (v_2, w_2)>_{\oplus} = <w_1, v_2> + <w_2, v_1> \qquad (1)$$

where $<v_i, w_i>$ *denotes the duality product of a pair* $(v_i, w_i) \in \mathcal{V} \times \mathcal{V}^*$.

Definition 2: Dirac structure on a vector space [5]

A *Dirac structure on a vector space* $\mathcal{V}$ *is a subspace* $L \subset \mathcal{V} \oplus \mathcal{V}^*$ *which is maximally isotropic under the plus pairing* $<,>_+$ *s.t.:*

$$<(x, y), (x', y')>_{\oplus} = \frac{1}{2} \left(<y, x'> + <y', x>\right) \qquad (2).$$

that is a subspace $L \subset \mathcal{V} \oplus \mathcal{V}^*$ *such that: i)* $\dim L = n$ *and: ii)* $\forall\ (a_1, a_2) \in L \times L,\ <a_1,\ a_2>_{\oplus} = 0$ *(3).*

It may be shown that the condition *ii)* is equivalent with the condition [14]:

$$\forall\ (v,w) \in L,\quad <w,v> = 0 \tag{4}$$

Practically, Dirac structures may be defined in a constructive way by using some particular basis $\mathfrak{B} = (b_1,\ ..\ ,\ b_n)$ of the vector space $\mathcal{V}$ and its dual basis $\mathfrak{B}^* = (b_1^*,\ ..\ ,\ b_n^*)$ of $\mathcal{V}^*$ using different linear representations [6] from which we recall the image representation.

Proposition 1: Image representation of a Dirac structure. [5]

A Dirac structure L on the real vector space $\mathcal{V}$ with dimension n is defined by 2 linear maps:

$a : \mathbb{R}^n \to \mathcal{V}$ *and* $b : \mathbb{R}^n \to \mathcal{V}^*$ *by:*

$$L = \operatorname{Im} a \oplus \operatorname{Im} b \tag{5}$$

where the two maps satisfy: $a^*b + b^*a = 0$ *and* $\ker a \cap \ker b = \{0\}$.

Furthermore, by taking the canonical basis of $\mathbb{R}^n$, denoted by $\mathfrak{B} = (e_1,\ ..\ ,\ e_n)$, the image representation defines the following basis of the Dirac structure L in $\mathcal{V} \oplus \mathcal{V}^*$: $\mathfrak{B}_L = \left((a(e_i)),\ (b(e_i))\right)_{i=1,..,n}$.

Dirac structures were derived in the case of electrical circuits where the represent the set of admissible currents and voltages [4] [11] or for spatial mechanisms where they represent the set of admissible twists and wrenches of the kinestatic model [10].

2. 2. Scattering variables on vector spaces

Consider a basis $\mathfrak{B}$ of $\mathcal{V}$ and its dual basis $\mathfrak{B}^*$ for $\mathcal{V}^*$, and composing the basis : $\mathfrak{B} \oplus \mathfrak{B}^* = \left((b_1,0),\ ..\ ,\ (b_n,0),\ \left((0,\ b_1^*),\ ..\ ,\ (0,\ b_n^*)\right)\right)$ for $\mathcal{V} \oplus \mathcal{V}^*$, and denoting by 0_n the null matrix and by I_n the identity matrix of order n, the canonical symmetric tensor (1) is defined by the matrix:

$$P = \begin{pmatrix} 0_n & I_n \\ I_n & 0_n \end{pmatrix} \tag{6}$$

It may be seen that the matrix admits eigenvalues *1* and *–1* with multiplicity *n* with eigenspaces S_+ (associated with 1) and S_-(associated with –1) which satisfy: $S_+ \oplus S_- = \mathcal{V} \oplus \mathcal{V}^*$. The space S_+ and S_- define the *subspaces of scattering variables*.

Definition 3: Subspaces of scattering variables

The subspaces of scattering variables are the two eigenspaces S_+ (associated with the eigenvalue 1) and S_- (associated with the eigenvalue –1) of the canonical symmetric tensor (1).

Restricting the bilinear form (1) to S_+, induces an *inner product on* S_+ which we denote by $<,>_+$ (and we denote by $\|.\|_+$ the corresponding norm). In the same way, restricting it to S_- induces (with a sign change) an *inner product on* S_- which we denote by $<,>_-$ (and we denote by $\|.\|_-$ the corresponding norm) [12] [13].

In the scattering representation, the symmetric tensor (1) is expressed as follows. Consider a pair of elements (v_1,w_1) and (v_2,w_2) in $\mathcal{V} \oplus \mathcal{V}^*$ and denote their decomposition in the scattering spaces by:

$$(v_i,w_i) = s_i^+ + s_i^-\ ,\quad i = 1,2 \tag{7}$$

where $s_i^+ = \varrho^+\left((v_i,w_i)\right)$ and ϱ^+ is the canonical projection on S_+ parallel to S_- and $s_i^- = \varrho^-\left((v_i,w_i)\right)$ and ϱ^- is the canonical projection on S_- parallel to S_+.

Then the symmetric tensor may be expressed in terms of the inner products on S_+ and S_- :

$$<(v_1,w_1),(v_2,w_2)>_{\oplus} = <\ s_1^+,\ s_2^+\ >_+ - <s_1^-,\ s_2^->_- \tag{8}.$$

In the case when the vector space corresponds to power variables of network models, for instance currents or voltages in an electrical circuit, this definition of scattering variables coincides with the usual definition (where, recall it again, the vector space and its dual are identified with $\mathbb{R}^n$ and the duality product is identified with the standard inner product) [2].

2. 3. Scattering representation of Dirac structures

Here we briefly recall the derivation of the scattering representation of a Dirac structure L proposed in [12] [13]. This derivation follows closely Courant's procedure in [5], however in a more intrinsic way as it does not use of any identification of the vector space $\mathcal{V}$ with $\mathbb{R}^n$ nor choose any inner product but solely the duality product.

Note firstly that the Dirac structure L is a vector subspace transversal to S_+ and S_- that is:

$L_{\mathfrak{B}} \cap S_+ = L_{\mathfrak{B}} \cap S_- = \{(0,0)\}$ as, according to (4), $\forall (v,\ w) \in L: \left((v,\ w),(v,\ w)\right)_{\oplus} = 0 \notin \{-1,1\}$.

Hence the restriction ϱ_L^+ and ϱ_L^- of the projection ϱ^+ and ϱ^- to L are *isomorphisms* and the Dirac structure L may as well be defined as the graph of an isomorphism from S_+ to S_-.

Definition 4: scattering representation of a Dirac structure

The scattering representation of a Dirac structure L *is the map* $\mathcal{O}$ *from* S_+ *to* S_-:

$$\mathcal{O} = \varrho_L^- \circ \varrho_L^{+-1} \tag{9}$$

where ϱ_L^+ and ϱ_L^- are the restriction to L of the projection ϱ^+ and ϱ^-.

Now consider an element $s^+ \in S_+$ and denote by $\lambda \in L$ its image by $\varrho_L^{+^{-1}}$. Then, by (8), $\|\mathcal{O}(s^+)\|_-^2 - \|(s^+)\|_+^2 = <\lambda,\ \lambda>_\oplus$ and as $\lambda \in L$, the scattering representation preserves the norms i.e.:

$$\|\mathcal{O}(s^+)\|_-^2 - \|(s^+)\|_+^2 = 0 \qquad (10).$$

Proposition 2:

The scattering representation $\mathcal{O}$ of a Dirac structure L is an isometry from S_+ to S_- endowed with the norms $\|,\|_+$ and $\|,\|_-$ respectively.

The scattering representation of Dirac structures associated with LC circuits was treated extensively in [4] [11], including LC circuits with elements in excess as well as the scattering representation of non constant pseudo–Poisson brackets [12].

3. Scattering variables for mechanisms

3. 1. Scattering variables associated with twists and wrenches

Let us first recall briefly the definition of the variables defining the kinestatic model of a mechanisms [10] [16]. These are of course the velocities of the bodies and the relative velocities for expressing the constraints induced by the kinematic pairs as well as the forces applied to the bodies or transmitted in the mechanism. However, in order to be able to express the kinestatic model of a mechanisms the velocities and forces have to be translated in some common references using left or right translation and are then called *twists* and *wrenches* [10] [16].

Definition 5: Twists in body frame

Let $Q(t) \in SE(3)$ be a trajectory of a rigid body (where $SE(3)$ denotes the Special Euclidean group in the three–dimensional Euclidean space), then the twist in body frame *is the image of the velocity by a linear map, called the tangent map to the left translation, denoted by $TL_{Q^{-1}}$:* $T = TL_{Q^{-1}}\left(\frac{dQ}{dt}\right)$ *(11)*

where L_{Q_0} denotes the left translation by Q_0 which maps $Q \in SE(3)$ into $L_{Q_0}(Q) = Q_0\ Q$.

Twist in body framebelong to the Lie algebra $se(3)$ of the group of rigid body displacements $SE(3)$. One may also define *twists in fixed frame* which are defined in a similar manner using the right translation and consider translations of twists by right or left multiplication by some displacements [16]. Twists with respect to different frames (for instance a body frame) are related through a linear map, called the *adjoint representation* [10] [16].

Definition 6: Adjoint representation

The adjoint representation, denoted by Ad_Q where $Q \in SE(3)$ is some rigid body displacement, is the tangent map to the map from $SE(3)$ on $SE(3)$: $P \mapsto Q\ P\ Q^{-1}$ *(12).*

In the same way forces are expressed as *wrenches* in fixed frame or in body frame (i.e. members of the dual Lie algebra $se^*(3)$) which are related by the adjoint map to the adjoint representation, denoted by Ad_Q^* [10] [16].

Now, as the set of twists in $se(3)$ is obviously a real vector space, one could define scattering variables extending the definition of electrical circuits by using an identification of $se(3)$ with $\mathbb{R}^6$ and using an arbitrary inner product [5]. However, taking into account that the twists and wrenches of a kinestatic model of a mechanisms are expressed in different frames, this identification should be invariant with respect to the adjoint representation Ad_Q. Furthermore one may show that there is no Ad_Q–invariant inner product on $se(3)$ ([16], chap.4). Hence one cannot use this definition in order to generalize scattering variables to twists and wrenches.

However, one may use the alternative definition proposed in section 2. 2. which uses only the duality product between the space of twists $se(3)$ and the space of wrenches $se^*(3)$. This duality product is obviously preserved under the composed map denoted by $\mathcal{A}_Q$ and transforming pairs (t, w) of twists and wrenches in $se(3) \oplus se^*(3)$ from one frame to the other by:

$$(t, w) \mapsto \mathcal{A}_Q(t, w) = \left(Ad_Q\ t,\ Ad^*_{Q^{-1}}\ w\right) \qquad (13).$$

Hence the scattering subspaces are also mapped into each other by the map $\mathcal{A}_Q$ which is, restricted to the scattering subspaces, also an isometry with respect to their inner products. Therefore the definition of the scattering variables is now invariant with respect to changes of frames, i.e. to the adjoint representation and the composed map $\mathcal{A}_Q$.

Proposition 3: scattering variables for spatial mechanisms

Scattering variables defined on the space $se(3)$ of twists, according to Definition 3, are invariant with respect to changes of frames, i.e. with respect to the adjoint representation Ad_Q

3. 2. Scattering representation of kinestatic models

The kinestatic model of a mechanism describes the admissible set of twists and wrenches of the rigid bodies and the spatial springs constituting a spatial mechanical system. This model may be decomposed in a network model as the interconnection of kinematic pairs through a port connection graph which define admissible sets of twists and wrenches as belonging to some Dirac structures [15].

The *port connection graph* of a mechanism is an oriented graph describing the topology of the mechanism and whose edges are associted with the pairs of twists and wrenches of the bodies, springs and kinematic pairs composing the mechanism [7] [10]. Its twists and wrenches may be regarded as across and through variables of this mechanical network like voltages and currents in an electrical network and it may be shown that obey *generalized Kirchhoff's laws* applied to the port interconnection graph [7]. Its scattering representation will not be recalled here but the reader is refered to [11].

A kinematic pair is the kinematic idealization of a set of contacts that occur between two rigid bodies. The wrench W transmitted by a kinematic pair is constrained to a linear subspace of the space of wrenches $se^*(3)$ called the *space of constraint wrenches* and denoted by $\mathcal{CW}$ of dimension $6-d$ where d is the number of degrees of freedom of the kinematic pair. A relative twist between the two bodies is allowed by the kinematic pair when it produces no work with any transmissible wrench. The relative twist is thus constrained to belong to a linear subspace of the screw vector space, called the *space of freedom twists* and denoted by $\mathcal{FT}$ which is orthogonal, in the sense of the duality product, to the space of constraint wrenches $\mathcal{CW}$. Hence the admissible sets of twists and wrenches at a kinematic pair form the Dirac structure: $\mathcal{FT} \oplus \mathcal{CW}$. Using a basis adapted to the spaces of constraint wrenches and freedom twists [3], one may then show that the scattering representation is then in a matrix representation: $O = \mathrm{diag}(I_d, -I_{6-d})$ which indicates that the scattering variable is transmitted along the degrees of freedom and reflected along the constraints.

References

[1] R.J. Anderson and M.W. Spong, "Asymptotic stability for force reflecting teleoperators with time delay", *The Int. J. of Robotics Research*, Vol.11, No. 2, pp..135–149, 1992

[2] V.Belevitch, *Classical Network Theory*, Holden–Day, San Francisco, 1968

[3] C. Bidard, "Dual bases of screw–vectors for inverse kinestatic problems in robotics", soumis au 4th Int. Workshop on Advances in Robot Kinematics, Ljubljana, July 1994

[4] A.M.Bloch and P.E.Crouch, "Representation of Dirac structures on Vector Spaces and Nonlinear L–C Circuits", *Proceedings of Symposia in Pure Mathematics, Differential Geometry and Control Theory*, G. Ferreyra, R. Gardner, H. Hermes, H.Sussmann, eds. Volume 64, pp. 103 – 117, A.M.S., 1999.

[5] T.J.Courant, "Dirac manifolds", *Trans. American Mathematical Society*, Vol. 319, n°2, pp.631–661, June 1990

[6] M.Dalsmo and A.J.van der Schaft, "On the representations and integrability of mathematical structures in energy conserving physical systems", *SIAM J. on Control and Optimization*, Vol. 37, No 1, pp.54–91, 1999.

[7] T. H. Davies, "Kirchhoff's circulation law applied to multi–loop kinematic chains", *Mechanism and Machine Theory*, vol. 16, 171–183, 1981

[8] I.Dorfman, *Dirac Structures and Integrability of Nonlinear Evolution Equations*, John Wiley,Chichester, 1993

[9] Maschke B.M., van der Schaft A. and Breedveld P.C., "An Intrinsic Hamiltonian Formulation of Network Dynamics: Nonstandard Poisson Structures and Gyrators", *Journal of the Franklin Institute*, Vol. 329, n. 5, pp. 923–966, 1992

[10] B.M.J.Maschke and A.J. van der Schaft, "Interconnected mechanical systems. Part 2: The dynamics of spatial mechanical networks" in *Modelling and Control of Mechanical Systems*, A.Astolfi, D.J.N.Limebeer, C.Melchiorri, A. Tornambè and R.B.Vinter eds., pp.17–30, Imperial College Press, 1997

[11] B.M.Maschke and A.J. van der Schaft, "*Scattering representation of Dirac structrures and interconnection in network models*", Proc. Int. Conf. on Mathematical Theory of Networks and Systems MTNS'98, A.Beghi, L. Finesso, G.Picci eds., pp. 305–308, Il Poligrafo, Padova, Italy, 1998

[12] B.M.Maschke et A.J. van der Schaft, "Hamiltonian systems, pseudo–Poisson brackets and their scattering representation for physical systems", *CD Proc. Int. Symp. on Motion and Vibration Control, DETC99/VIB–8007, 17th ASME Biennal Conf. on Mechanical Vibation and Noise*, Las Vegas, Nevada, Sept. 12–15, 1999

[13] A.J. van der Schaft, *L_2–gain and Passivity Techniques in Nonlinear Control*, Communications and control engineering series, Springer, London, 2000

[14] A.J. van der Schaft and B.M.Maschke, "The Hamiltonian formulation of energy conserving physical systems with ports", *Archiv für Elektronik und Übertragungstechnik*, Vol.49, n°5/6, pp.362–371, 1995.

[15] A.J. van der Schaft and B.M.J.Maschke, "Interconnected mechanical systems. Part 1: Geometry of interconnection and implicit Hamiltonian systems" in *Modelling and Control of Mechanical Systems*, A.Astolfi, D.J.N.Limebeer, C.Melchiorri, A. Tornambè and R.B.Vinter eds., pp. 1–16, Imperial College Press, 1997

[16] J. M. Selig, Geometrical Methods in Robotics, Monographs in Computer Sciences, Springer, New–York, 1996

ON THE BLOCK STRUCTURE OF J-INNER FUNCTIONS

B. Kirstein[1] and K. Müller[2]
[1]Mathematical Institute, Leipzig University
Augustusplatz 1, D 04109 Leipzig
[2]Max-Planck Institute of Cognitive Neuroscience
Stephanstraße 1a, D 04103 Leipzig

Abstract. This paper is aimed at analyzing the canonical block structure of a special class of J-inner functions. Inspired by papers of Arov [1] and Dewilde/Dym [6], [7] a concept of parametrization of such functions is developed.

1 Introduction

The class of J-inner functions turned out to play an important role in the framework of matricial generalizations of classical interpolation problems of Schur-Nevanlinna-Pick type. Namely, the set of solutions of such an interpolation problem can be parametrized with the aid of linear fractional transformations the generating matrix-valued functions of which are certain J-inner functions appropriately constructed from the given data (see [3], [5] and [8]). The inverse question of constructing interpolation problems such that their solution sets can be parametrized by a given function in the above described way, was also studied in [3]. Inverse problems for J-inner functions with prescribed block information are the content of the papers [1] and [4].

In this paper we are looking for appropriate parametrizations of j_{qq}-inner functions. The concept is based on ideas going back to Arov [2] and Dewilde/Dym [6], [7], where (without explicitly mentioning this) the special case of j_{qq}-inner functions belonging to the Smirnov class was treated. For this reason, the parametrization worked out in Section 5 is called the ADD-parametrization. A remarkable feature of this parametrization is the fact that a j_{qq}-inner function can be described by a connected pair of matrix-valued functions which belong to the Hardy class and another function which belongs to the Carathéodory class. Moreover, a procedure of constructing j_{qq}-inner functions is presented with prescribed ADD-parameters.

2 On functions of the meromorphic Nevanlinna class

In the first section we will summarize some facts on several classes of meromorphic functions. Throughout this paper, let p and q be positive integers. We will use C, D, T, C_0 and E to denote the set of complex numbers, the open unit disc, the unit circle, the extended complex plane and the exterior of the closed unit disc.

A matrix A is called *contractive*, respectively, *strictly contractive*, if $I - A^*A$ is nonnegative Hermitian, respectively, positive Hermitian. The linear Lebesgue-Borel measure on the unit circle will be designated by $\underline{\lambda}$.

Assume that G is a simply connected domain of the extended complex plane. Then let $\mathcal{NM}(G)$ be the *Nevanlinna class* of all functions which are meromorphic in G and which can be represented as a quotient of two bounded holomorphic functions in G. Note that the well-known Hardy classes are subsets of the meromorphic Nevanlinna class. If G is the open unit disc or the exterior of the unit disc, and if $g \in \mathcal{NM}(G)$, then a well-known theorem due to Fatou implies that there exists a radial boundary function $\underline{g}$ on the unit circle T.

Now we will introduce the most important concept of this paper. Let g be a function of $\mathcal{NM}(D)$. Then one says that g *admits a pseudocontinuation (into E)* if there exists a function $g^{\#}$ of the meromorphic Nevanlinna class $\mathcal{NM}(E)$ such that the radial boundary values $\underline{g}$ and $\underline{g^{\#}}$ coincide $\underline{\lambda}$-almost everywhere on T. It is obvious that a function of the meromorphic Nevanlinna class admits at most one pseudocontinuation. Note that if g admits a pseudocontinuation $g^{\#}$ and if, additionally, g is analytically continuable through some open arc of the unit circle T, then the analytic continuation coincides with the pseudocontinuation. In the sequel, we will continue to write $g^{\#}$ for the pseudocontinuation of g.

We will also use the following notion for a matrix-valued function f. For all elements z of the extended complex plane where $f(1/\overline{z})$ is defined, we will use the symbol $\widehat{f}$ to denote the function $\widehat{f}(z) = f(1/\overline{z})^*$.

3 About Schur and Carathéodory functions

In this section, we will deal with special classes of matrix-valued functions which are of interest for further considerations. A matrix valued function is said to be a *matrix-valued Schur function* if it is both holomorphic and contractive in the open unit disc. The set of all matrix-valued Schur functions is obviously a subset of the Hardy class of all bounded holomorphic matrix-valued functions. A Schur function is called an *inner function* if it has unitary radial boundary values $\underline{\lambda}$-almost everywhere on the unit circle. If it has even strictly contractive values in D, then it is said to be a *strictly contractive Schur function.*

A matrix-valued function is said to be a *matrix-valued Carathéodory function* if it is holomorphic and if its real part is nonnegative Hermitian in the open unit disc. This set of functions is a subset of the meromorphic Nevanlinna class. In particular, every matrix-valued Carathéodory function has radial boundary values with nonnegative Hermitian real part $\underline{\lambda}$-almost everywhere on the unit circle.

For functions Ω of the Carathéodory class, the matricial version of a famous theorem due to F. Riesz and Herglotz provides that there is a unique nonnegative Hermitian-valued Borel measure F on the unit circle T such that

$$\Omega(w) = \int_T \frac{z+w}{z-w} F(\mathrm{d}z) + i \operatorname{Im}[\Omega(0)] \tag{1}$$

is satisfied. This nonnegative Hermitian-valued measure F is called the *F. Riesz-Herglotz measure associated with* Ω. A Carathéodory function Ω is said to be *absolutely continuous*, respectively, *singular* if the F. Riesz-Herglotz measure associated with Ω is absolutely continuous, respectively, singular, with respect to the linear Lebesgue-Borel measure $\underline{\lambda}$ on the unit circle. Note that every singular Carathéodory function admits a pseudocontinuation. The real part of a radial boundary value of a singular Carathéodory function is zero $\underline{\lambda}$-almost everywhere on the unit circle T.

If Σ is a Lebesgue-integrable function such that $\Sigma(z)$ is nonnegative Hermitian for $\underline{\lambda}$-almost all $z \in T$, then

$$\Omega_\Sigma(w) := \frac{1}{2\pi} \int_T \frac{z+w}{z-w} \Sigma(z) \underline{\lambda}(\mathrm{d}z) \tag{2}$$

is an absolutely continuous Carathéodory function. The real part of a radial boundary function of Ω_Σ coincides with Σ $\underline{\lambda}$-almost everywhere on T. If Σ and Ξ are Lebesgue-integrable functions, which have nonnegative Hermitian values $\underline{\lambda}$-almost everywhere on T and which coincide $\underline{\lambda}$-almost everywhere on T, then also the functions Ω_Σ and Ω_Ξ given by (2) coincide, i.e., the definition of (2) depends only on the equivalence class $\langle\Sigma\rangle$ of all functions Ξ which coincide with Σ $\underline{\lambda}$-almost everywhere on the unit circle.

In the sequel, we are interested in the special subclass of the Carathéodory class. We will denote by $C_{\langle\Sigma\rangle}$ the subset of all Carathéodory functions with an absolutely continuous part pre-determined by an Lebesgue-integrable nonnegative Hermitian function Σ via (2). Each element of the set $C_{\langle\Sigma\rangle}$ can be represented as the sum $\Omega_{\langle\Sigma\rangle} + \Omega_s$ with a singular Carathéodory function Ω_s.

If ϕ is a function of the Hardy class and the radial boundary value $\underline{\phi}\underline{\phi}^*$ is Lebesgue-integrable, then the set $C_{\langle\underline{\phi}\underline{\phi}^*\rangle}$ is called the *subclass of the Carathéodory class left generated by* ϕ. Note, if ϕ admits furthermore a pseudocontinuation, then every function of the Carathéodory class left generated by ϕ admits a pseudocontinuation.

4 Some remarks on J-inner functions

Let J be a signature matrix, i.e., J is a quadratic complex matrix and satisfies as well $J = J^*$ as $J^2 = I$. A matrix A is called *J-contractive* if $J - A^*JA$ is nonnegative Hermitian. If A even satisfies $A^*JA = J$, then A is said to be *J-unitary.* The *Potapov class* $\mathcal{P}_J(D)$ consists of all matrix-valued meromorphic functions W with J-contractive values $W(z)$ for all points z of analyticity of W. The Potapov class $\mathcal{P}_J(D)$ is a subclass of the meromorphic Nevanlinna class. In particular, every function of the Potapov class has radial boundary values $\underline{W}$ almost everywhere on the unit circle T. If a Potapov function has almost everywhere J-unitary boundary values on the unit circle, then it is said to be a *J-inner function.* Note that every J-inner function admits a pseudocontinuation.

In the following we will focus our attention to the special signature matrix $j_{qq} := \operatorname{diag}(I_q, -I_q)$. When we will consider a $2q \times 2q$ complex matrix or a matrix-valued function W, then we will work with the block partition

$$W = \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix} \tag{3}$$

where each block of W has the size $q \times q$.

A useful tool to treat problems which are formulated for functions from the Potapov class is the so-called *Potapov-Ginzburg transform* which enables us to work in the Schur class. We will now briefly explain this. Let $W \in \mathcal{P}_{j_{qq}}(D)$. For all points z of analyticity of W, the inequalities

$$\det W_{22}(z) \neq 0\,, \quad \det[W_{22}(z) + W_{21}(z)] \neq 0 \quad \text{and} \quad \det[W_{22}(z) + W_{12}(z)] \neq 0 \tag{4}$$

hold true. The functions

$$\det(\widehat{W_{11}^{\#}} + \widehat{W_{12}^{\#}}) \quad \text{and} \quad \det(\widehat{W_{11}^{\#}} + \widehat{W_{21}^{\#}}) \tag{5}$$

do not identically vanish. The Potapov-Ginzburg transform

$$S := \begin{bmatrix} W_{11} - W_{12}W_{22}^{-1}W_{21} & W_{12}W_{22}^{-1} \\ -W_{22}^{-1}W_{21} & W_{22}^{-1} \end{bmatrix} \tag{6}$$

of W belongs to the Schur class. If S is partitioned into $q \times q$ blocks, then its blocks are matrix-valued Schur functions, whereby the functions S_{12} and S_{21} on the secondary diagonal are even strictly contractive. The functions $\det S_{11}$ and $\det S_{22}$ do not identically vanish. If W is additionally a j_{qq}-inner function, then S is an inner function and the following identities are valid:

$$S_{12} = (\widehat{W_{11}^{\#}})^{-1}\widehat{W_{21}^{\#}}\,, \; S_{21} = -\widehat{W_{12}^{\#}}(\widehat{W_{11}^{\#}})^{-1} \text{ and } S_{11} = (\widehat{W_{11}^{\#}})^{-1}. \tag{7}$$

5 A parametrization of j_{qq}-inner functions

For the announced parametrization, some concept of association between matrix-valued functions of the Hardy class is used (see [9]). An ordered pair (ϕ, ψ) of Hardy functions is called *left connected* if there is a matrix-valued inner Schur function v such that $\underline{\psi} = \underline{v}\underline{\phi}^*$ holds true $\underline{\lambda}$-almost everywhere on the unit circle T. Every such function v is said to be an *inner function which realizes the left connection between ϕ and ψ*.

If W is a j_{qq}-inner function, then the pair (Φ_W, Ψ_W) given by

$$\Phi_W := (W_{22} + W_{21})^{-1} \quad \text{and} \quad \Psi_W := (\widehat{W_{11}^{\#}} + \widehat{W_{12}^{\#}})^{-1} \tag{8}$$

is a left connected pair of Hardy functions which is also called *the left connected pair generated by W*. The unique inner matrix-valued Schur function

$$V_W := (W_{11} + W_{12})(W_{22} + W_{21})^{-1} \tag{9}$$

realizes the left connection between Φ_W and Ψ_W. Note that both functions Φ_W and Ψ_W admit pseudo-continuations and satisfy the identities

$$V_W\widehat{\Phi_W^{\#}} = \Psi_W \quad \text{and} \quad \widehat{\Psi_W^{\#}}V_W = \Phi_W\,. \tag{10}$$

The Carathéodory function contructed via

$$\Omega_W := (W_{22} + W_{21})^{-1}(W_{22} - W_{21}) \tag{11}$$

belongs to the subclass which is left generated by Φ_W. This function is called *the Carathéodory function left generated by W*. Note that the absolutely continuous part of Ω_W is uniquely determined by Φ_W, i.e., the function $\Omega_{W;s} := \Omega_W - \Omega_{\Phi_W}$ is a singular Carathéodory function. The j_{qq}-inner function W admits the representation

$$W = \frac{1}{2}\operatorname{diag}((\widehat{\Psi_W^{\#}})^{-1}, \Phi_W^{-1}) \begin{bmatrix} I + \widehat{\Omega_W^{\#}} & I - \widehat{\Omega_W^{\#}} \\ I - \Omega_W & I + \Omega_W \end{bmatrix}. \tag{12}$$

This representation is unique, i.e., if there is a further left connected pair and a further Carathéodory function realizing that representation, then the left connected pair coincides with (Φ_W, Ψ_W) and the Carathéodory function with Ω_W. The triple $(\Phi_W, \Psi_W, \Omega_{W;s})$ is called the *left ADD-parametrization of the j_{qq}-inner function W.*

Now we will turn our attention to the inverse question, namely to construct j_{qq}-inner functions with prescribed ADD-parameters. Let (Φ, Ψ) be a left connected pair of Hardy functions such that the function $\det \Phi$ does not identically vanish. Further, let Ω be a function of the Carathéodory subclass left generated by Φ. Then Ω admits a pseudocontinuation and

$$W := \frac{1}{2} \operatorname{diag}((\widehat{\Psi^{\#}})^{-1}, \Phi^{-1}) \begin{bmatrix} I + \widehat{\Omega^{\#}} & I - \widehat{\Omega^{\#}} \\ I - \Omega & I + \Omega \end{bmatrix} \tag{13}$$

is a j_{qq}-inner function. Moreover, (Φ, Ψ) and Ω are the left connected pair of Hardy functions and the left Carathéodory function, respectively, generated by W. If V is the (unique) inner Schur function which realizes the left connection of (Φ, Ψ), then

$$S = \begin{bmatrix} 2\Psi(I+\Omega)^{-1} & V - 2\Psi(I+\Omega)^{-1}\Phi \\ -(I-\Omega)(I+\Omega)^{-1} & 2(I+\Omega)^{-1}\Phi \end{bmatrix} . \tag{14}$$

is a matrix-valued inner Schur function. The representation (14) is the Potapov-Ginzburg transform of the j_{qq}-inner function W.

The following modification of the result given above provides now a complete answer to the inverse question associated with ADD-parametrization. Namely, let (Φ, Ψ) be a left connected pair of Hardy functions such that the function $\det \Phi$ does not identically vanish, and let Ω_s be a singular Carathéodory function. Then there is a unique j_{qq}-inner function W such that (Φ, Ψ, Ω_s) is the left ADD-parametrization of W.

References

[1] Arov, D. Z.: *Darlington realization of matrix-valued functions* (in Russian), Izv. Akad. Nauk SSSR, Ser. Mat. **37** (1973), 1299–1331; Math. USSR Izvestija **7** (1973), 1295–1326.

[2] Arov, D. Z.: *On functions of class* Π (in Russian), Zap. Nauc. Sem. LOMI **135** (1984), 5–30.

[3] Arov, D. Z.: *$\gamma-$generating matrices, $J-$inner matrix-functions and related extrapolation problems* (in Russian), Teor. Funkcii, Funk. Anal. i Prilozen., Part I: **51** (1989), 61–67; Part II: **52** (1989), 103–109; Part III: **53** (1990), 57–64; J. Soviet Math. **52** (1990), 3487–3491; **52** (1990), 3421–3425; **58** (1992), 532–537.

[4] Arov, D. Z.; Fritzsche, B.; Kirstein, B.: *Completion problems for $j_{pq}-$inner Functions*, Integral Equations and Operator Theory, I: **16** (1993), 155–185; II: **16** (1993), 453–495.

[5] Ball, J. A.; Gohberg, I.; Rodman, L.: *Interpolation of Rational Matrix Functions*, Operator Theory Series, Vol. 45, Birkhäuser, Basel 1990.

[6] Dewilde, P.; Dym, H.: *Schur recursion error formulas and convergence of rational estimations for stationary stochastic processes*, IEEE Trans. Inf. Theory **27** (1981), 416–461.

[7] Dewilde, P.; Dym, H.: *Lossless chain scattering matrices and optimum linear prediction: the vector case*, Intern. J. Circuit Theory Appl. **9** (1981), 135–175.

[8] Dym, H.: *J-contractive matrix functions, reproducing kernel Hilbert spaces and interpolation*, CBMS Regional Conf. Ser. Math. 71, Amer. Math. Soc., Providence, R. I. 1989.

[9] Fritzsche, B.; Kirstein, B.; Müller, K.: *An analysis of the block structrure of j_{qq}-inner functions*, Operator Theory: Advances and Applications, Vol. 106, Birkhäuser, Basel 1998, pp. 157–185.

DIFFERENTIAL GEOMETRIC MODELS FOR NONLINEAR MANIFOLDS AND MULTIPORT MODELS FOR LINEAR PHYSICAL SYSTEMS

R. Pauli
Munich University of Technology
D – 80290 Munich, Germany

Abstract. In order to break fresh ground in engineering multiport theory and to facilitate a cross-fertilization with current trends in mathematical systems theory, we sketch some clear-cut correspondences between the essentially coordinate-free black-box viewpoint in multiport theory and "modern" geometric concepts of mathematical systems theory. For this sake, we point out the role of Stiefel and Grassmann manifolds in linear multiport theory and disclose analogies between standard cascade models of multiport theory and models for manifolds in differential geometry (homogeneous spaces or group models and principal fiber bundles). Added physical constraints enter into this picture in terms of transformation groups that leave physically meaningful properties invariant. We stress the underlying ideas and the interrelations between engineering blackbox models and geometrical methods in mathematical systems theory rather than the geometric-topological techniques.

1 Motivation and Introduction

Since the pioneering work of R. Kalman, R. Brockett, M. Hazewinkel, R. Hermann, and C. Martin geometric and topological methods have found broad accceptance in *mathematical* systems and control theory [5]. However, there are only very few serious attempts to acquaint engineers with the basics of modern global differential geometry in general and with homogeneous spaces and special manifolds like Grassmannians in detail. Agreeable attempts of this type are for instance [11] or the numerous "digressions" in [12]. Strangely enough, though Grassmannians have entered into engineering *applications* via Riccati equations, pole placement, array signal processing and coding, the opportunity to introduce (at least electrical) engineers to such concepts in a most natural way via Grassmannians as sets of multiports [14] has widely been ignored.

It is in the nature of network synthesis as well as systems theory to look at *sets* or families of related objects. Exactly this is the reason why a profound knowledge of *linear* network and systems theory actually requires the study of the global properties of differentiable manifolds. This may be immediately recognized as follows: A concrete linear n-port is completely characterized (in the black-box sense) by the linear n-dimensional subspace spanned by all admissible signal pairs at the n ports. The set of all linear n-ports (corresponding to the set of n-dimensional subspaces of a linear vector space), however, is not a linear space! Rather this set shows the topological structure of a pretty nonlinear differentiable manifold that is called after Hermann Graßmann who first invented the concept of abstract vector spaces of dimension $n > 3$.

Grassmannian manifolds (or in short "Grassmannians") are fundamental objects in topology and algebraic as well as differential geometry. Simultaneously, they are objects of interest in their own right and basic tools in the construction and study of other differentiable manifolds. In much the same spirit engineers have been studying multiports in their own right and as basic building blocks for the construction of models for complex systems. Clearly, interconnection of basic building blocks must be performed according to strong rules in order to ensure meaningful results; this is true in topology (cell complexes [16]) as well as in multiport theory (where Kirchhoff's rules take the role in guaranteeing fulfillment of the physical constraints concerning conservation of energy in interconnected systems). For circuit theory, as well as for any other branch of physical systems engineering it is still true what R. E. Kalman wrote in 1963: "Control theory is supposed to deal with physical systems, and not merely with mathematical objects such as a differential equation or a transfer function. We must therefore pay careful attention to the relationship between physical systems and their representation via differential equations, transfer functions, etc."[15]. There should be no doubt that a systems theory aiming at physical reality, even in the seemingly trivial case of linear systems, must incorporate in one way or another the following "chief ingredients":

Notion of ports: Like currents and voltages in electrical circuits, signals in physical structures are goverened by a *natural pairing* of two types of variables: (i) flow or through variables and (ii) effort, pressure, or across variables. Any natural pair of variables is associated with a "port" in such a way that the power or energy flow at the port is uniquely determined by the scalar product of the pertaining dual signal pair as e.g., in the case of electrical power P in terms of a voltage/current pair $(u, i) \in \mathsf{R}^n \times \mathsf{R}^n$,

$$P = u^{\mathrm{T}} i = \frac{1}{2} \begin{bmatrix} u \\ i \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} 0_n & 1_n \\ 1_n & 0_n \end{bmatrix} \begin{bmatrix} u \\ i \end{bmatrix}. \tag{1}$$

As is well-known in multiport theory and in the closely related bondgraph concept, the latter condition is crucial for the correct modeling and engineering synthesis of physical systems: "Experience with electrical network synthesis suggests that the synthesis problem for natural pairs is the key to all synthesis problems"[3].

Bilinear forms: Fundamental physical properties of a multiport (such as passivity, losslessness or reciprocity) are related to forced conditions in terms of various bilinear forms, where the actual appearance of the form matrices or metric tensors depends on the choice of local coordinates. However, when combining natural pairs like u and i into a port vector $x = [u_1, \ldots, u_n, i_1, \ldots, i_n] \in \mathsf{R}^n \oplus \mathsf{R}^n$, this may be reduced to the elementary symmetric and anti-symmetric forms given (modulo congruence) by the $(2n \times 2n)$ matrices

$$J = \begin{bmatrix} 1_n & 0_n \\ 0_n & -1_n \end{bmatrix}, \quad H = \begin{bmatrix} 0_n & 1_n \\ -1_n & 0_n \end{bmatrix}. \tag{2}$$

Note that J is simply the principle axis version of the metric tensor in eq. (1).

Non-Euclidean geometries: In customary coordinates, energy flow corresponds to symmetric forms that may be positive, negative, or *indefinite* (depending of the direction of energy flow at the various ports). Hence, the signal spaces in physical systems are indefinite inner product spaces and reflect the basically non-euclidean geometric nature of physical systems (recall the once well-known non-euclidean structure of classical two-port theory over the field of complex numbers [22] and its extension to multiports over function spaces [13]). One should recognize that any formulation of intrinsically physical problems in an Euclidean setting is based on the ubiquitous identification of energy with a positive quadratic form and requires a *local* choice of coordinates and/or restrictive assumptions on the type of system and its external excitation.

Manifolds: The classical way in which manifold theory enters physical systems is in terms of state manifolds in configuration space, that is as solution sets of the descriptive differential-algebraic equations with *constant* parameters. Another way is to look at smooth manifolds in parameter space. This is exactly what has been implicitly done in classical network synthesis since the late twenties (although mostly without recourse to manifold theory). A similiar trend was started in mathematical systems theory in the mid-seventies and resulted in a strengthening of global analytic aspects in dynamical systems theory (cf. [2], [5], [12]).

Lie groups: There is no need to emphasize Lie groups and their fundamental role in physics [20]. In the present context they arise as the principal group in geometric-topological models for smooth manifolds, i.e. in terms of group models and fiber bundles [10], [21]. The connection with circuit theoretic black-box models arises exactly when one looks at Lie groups which preserve elementary symmetries of the object under study or leave invariant physically relevant bilinear forms. Most important are the pseudo-orthogonal group $\mathbf{O}_{n,n} = \{T \in \mathbf{GL}_{2n} \mid T^{\mathrm{T}} J T = J\}$ and the symplectic group $\mathbf{Sp}_n = \{T \in \mathbf{GL}_{2n} \mid T^{\mathrm{T}} H T = H\}$ that are defined by the invariance of the bilinear forms given by eq. (2).

2 Stiefel Manifolds and Grassmannians

Stiefel manifolds and Grassmannians occur quite naturally in multiport theory. Though these manifolds are well defined over any commutative field with a unit, in order to enucleate the most elementary structures, we limit ourselves to resistive multiports and hence to the real numbers R as a ground field. This is also justified by the fact that physical systems are governed by real parameters. In case of dynamical multiports one may take the complex field C for the spot-frequency behavior or real function spaces over C [23], [14]. Any linear, source-free n-port is uniquely defined in the black-box sense by its external behavior, that is (excluding unphysical degenerations) by the linear n-dimensional subspace of all admissible signal pairs in $2n$-space. This may be conceived by a point on the Grassmannian $\mathbf{Gr}(n, \mathsf{R}^{2n})$ or in short, $\mathbf{Gr}_n$. Hence, the set of all linear (time-invariant, resistive) n-ports may be identified with $\mathbf{Gr}_n$. How does this abstract concept match with engineering multiport theory?

According to Youla's "Q-matrix method" [23] the most natural engineering description of an n-port is by n vector valued measurements of linearly independent admissible signal pairs $(u_i, i_i) := x_i$. These pairs are collected together as the columns of a basis matrix (n-frame) $X = [x_1, x_2, \ldots x_n] \in \mathbf{St}_n$, the noncompact Stiefel manifold of $(2n \times n)$ matrices of full rank n. Any other set of measurements simply results in a change of basis $X \mapsto XT$, $T \in \mathbf{GL}_n$, where $\mathbf{GL}_n$ is the general linear group of invertible $(n \times n)$ matrices. In other words, any two n-frames X, Y are equivalent (i.e., obtained from different measurements of the same n-port) precisely when they span the same space:

$$X \sim Y \iff \mathrm{span}X = \mathrm{span}Y \iff Y = XT, \qquad X, Y \in \mathbf{St}_n,\ T \in \mathbf{GL}_n\,.$$

In essence, this engineering description of n-ports as equivalence classes of measurements corresponds to the conception of $\mathbf{St}_n$ as a $\mathbf{GL}_n$ principal fiber bundle over $\mathbf{Gr}_n$, where the canonical bundle projection $\pi : \mathbf{St}_n \to \mathbf{Gr}_n$ amounts to the identification of a concrete n-port (given by a "fiber" of measurements) with an abstract point on the Grassmannian $\mathbf{Gr}_n$.

A most important point is the compactness of Grassmannians – otherwise synthesis, identification or approximation of multiports would be much a more difficult task than it actually is. Any linear n-port may be represented by an orthogonal basis $X \in \mathbf{St}_n$ modulo orthogonal changes of basis $X \mapsto XT$, $T \in \mathbf{O}_n$, where $\mathbf{O}_n$ is the group of orthogonal $(n \times n)$ matrices. The fact that $\mathbf{O}_{2n}$ acts transitively from the left on orthogonal n-frames and n-spaces results in the well-known differential geometric model $\mathbf{Gr}_n \cong \mathbf{O}_{2n}/(\mathbf{O}_n \times \mathbf{O}_n)$ that exhibits $\mathbf{Gr}_n$ as a homogeneous space of the compact group $\mathbf{O}_{2n}$ and hence as a compact manifold.

The best numerical way of representing n-ports is by use of orthogonal bases. It is interesting to observe that this fact has not been really appreciated in engineering circuit theory. The standard prescriptions of how to exploit the properties of multiports by way of external measurements inevitably result in non-orthogonal bases (except for trivial degenerate multiports). Engineers traditionally prefer to work with matrix representations as may be seen here in the impedance matrix Z and the admittance matrix Y:

$$\pi : \mathbf{St}_n \to \mathbf{Gr}_n, \quad X = \begin{bmatrix} U \\ I \end{bmatrix} = \begin{bmatrix} Z \\ 1_n \end{bmatrix} I = \begin{bmatrix} 1_n \\ Y \end{bmatrix} U \quad \mapsto \quad \operatorname{span}\begin{bmatrix} U \\ I \end{bmatrix} = \operatorname{span}\begin{bmatrix} Z \\ 1_n \end{bmatrix} = \operatorname{span}\begin{bmatrix} 1_n \\ Y \end{bmatrix}.$$

It follows that the representation of an n-port by $(n \times n)$ matrices amounts to the introduction of affine coordinates on $\mathbf{Gr}_n$ [14], [19]. For each matrix representation like $Z = UI^{-1}$ or $Y = IU^{-1}$ a specific $(n \times n)$ submatrix built from n "controlled" rows of the basis matrix X is required to have a non-zero determinant (e.g. $\det U \neq 0$ for an admittance matrix Y). Consequently, the set of all n-ports with a specific matrix representation is an open subset of $\mathbf{Gr}_n$. Let a multi-index $\alpha = (\alpha_1, \ldots, \alpha_n)$, $1 \leq \alpha_1 < \ldots < \alpha_n \leq 2n$, point on these n constrained rows and denote by $\bar{\alpha}$ its complement in $(1, \ldots, 2n)$. This way, a *standard chart* $(\mathbf{U}_{\bar{\alpha}}, M)$ is determined on $\mathbf{Gr}_n$, where the indices of $\bar{\alpha}$ point on the n "free" rows of a basis matrix $X \in \mathbf{St}_n$. The neighborhood $\mathbf{U}_{\bar{\alpha}}$ is the set of all n-planes in R^{2n} which allow for a representation as the graph of some $(n \times n)$ matrix M, i.e. (after some permutation of indices $(1, \ldots, 2n)$ such that the α_i are coming first and the $\bar{\alpha}_i$ at last),

$$\operatorname{graph}(M) = \{(x, Mx) \,|\, x \in \mathsf{R}^n\} = \operatorname{span}\begin{bmatrix} 1_n \\ M \end{bmatrix} \in \mathbf{Gr}_n .$$

The points "at infinity" of this euclidean set correspond to n-ports which do not allow for this specific matrix representation. In case of an admittance description, all points of $\mathbf{Gr}_n$ given by the condition $\det U = 0$ are excluded from this set (in circuit theoretic terms it corresponds to the set of not-completely-voltage-controlled n-ports). Circuit engineers circumvent difficulties with non-existence of a specific matrix representation simply by switching to another one, as e.g. for $n = 2$:

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ a & b \\ c & d \end{bmatrix} \quad \begin{bmatrix} 1 & 0 \\ a & b \\ 0 & 1 \\ c & d \end{bmatrix} \quad \begin{bmatrix} 1 & 0 \\ a & b \\ c & d \\ 0 & 1 \end{bmatrix} \quad \begin{bmatrix} a & b \\ 1 & 0 \\ 0 & 1 \\ c & d \end{bmatrix} \quad \begin{bmatrix} a & b \\ 1 & 0 \\ c & d \\ 0 & 1 \end{bmatrix} \quad \begin{bmatrix} a & b \\ c & d \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Here, a, b, c, d are the entries of any specific matrix representation. In general, there are $\binom{2n}{n}$ different open Euclidean sets (charts) of this kind. Invoking the picture of a smooth manifold as the result of glueing together floppy pieces of Euclidean space, it should be at least intuitively clear that they form a differentiable atlas of charts for $\mathbf{Gr}_n$. In other words: Electrical engineers are working with overlapping parametrizations of $\mathbf{Gr}_n$ by standard charts. Non-overlapping decompositions of $\mathbf{Gr}_n$ occur implicitly, e.g. in the case of the derivation of ideal transformer multiports as a limiting case of purely resistive (non-inductive) multiports [6]. The circuit-theoretic role of various (Schubert) cell decompositions or various stratifications of the Grassmannian in seems not to have been investigated thoroughly.

So far we have only inspected the consequence of the linearity assumption for the structural properties of sets of multiports. Added physical properties most often result in restrictions due to added symmetries in the external behavior. Lossless or reciprocal multiports, for instance, are characterized as totally isotropic subspaces $\operatorname{span} X$ with respect to an appropriate bilinear form; i.e., for the gramian matrices of any basis $X \in \mathbf{St}_n$ we have with the metrics given in eq. (2), $X^{\mathrm{T}} J X = 0_n$ or $X^{\mathrm{T}} H X = 0_n$, respectively. Such restrictions define subsets of $\mathbf{Gr}_n$ or even submanifolds as in the well-known case of the Lagrange Grassmannian $\mathbf{LGr}_n \cong (\mathbf{O}_{2n} \cap \mathbf{Sp}_n)/\mathbf{O}_n \cong \mathbf{U}_n/\mathbf{O}_n$ which corresponds to the set of linear reciprocal multiports (e.g., [4]).

3 Group Actions and Multiport Cascade Circuits

Classical circuit synthesis aims at the creation, identification and equivalence transformation of internal models for the external behavior of black-box multiports. After some re-interpretation most synthesis procedures appear as a systematic application of a sequence of group actions in order to transform a given multiport into an elementary one within the same group orbit. Fig. 1 gives a sketch of this: Starting with an n-port M (that belongs to a certain class $\mathbf{M}$) one "extracts" a $2n$-port $G \in \mathbf{G}$ and ends up with a remainder n-port $Z \in \mathbf{M}$. $\mathbf{G}$ is any subgroup of $\mathbf{GL}_{2n}$ acting transitively on $\mathbf{M}$ (e.g. $\mathbf{O}_{n,n}$ in case of lossless $2n$-ports). Clearly, such a realization is highly non-unique due to the existence of $2n$-ports belonging to a special subclass $\mathbf{H} \subset \mathbf{G}$ defined by the invariance of Z. Assume Z to be given by a basis $X_Z \in \mathbf{St}_n$, then $\mathbf{H}$ is the stabilizer subgroup for the subspace $\mathrm{span} X_Z \in \mathbf{Gr}_n$. Fixing Z while scanning through all $G \in \mathbf{G}$ (by tuning the parameters appropriately) one gets a parametrization of the entire class $\mathbf{M}$ as the $\mathbf{G}$-orbit of Z; in other words, $\mathbf{M}$ is the set of all n-ports that admit a realization by connecting a fixed Z to the appropriate n ports of some $G \in \mathbf{G}$. Since group elements correspond to elementary multiport "sections", circuit synthesis is mainly focused on the groups $\mathbf{O}_{n,n}$ and/or $\mathbf{Sp}_n$ which leave invariant the standard realizability criteria related to passivity, losslessness or reciprocity. In essence, there is a bijection between $\mathbf{M}$ and $\mathbf{G}$ modulo actions of the stabilizer $\mathbf{H}$ – a bijection that has been exploited (using a different terminology) by electrical engineers for the purpose of modeling physical systems and stationary stochastic processes, as well as for the design of matrix algorithms in terms of Lie group actions on a Grassmannian of subspaces (e.g. [9]).

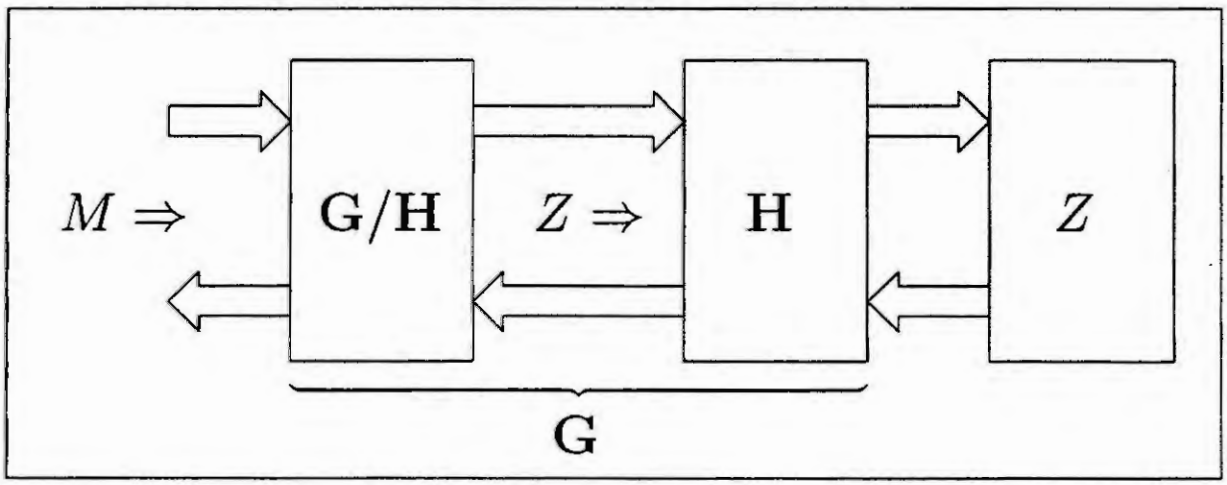


Fig. 1. Black Box cascade model.

Sets of multiports generated in this way are smooth manifolds. In manifold theory, however, it is well known that taking the quotient of a manifold by any equivalence relation usually leads out of the category of manifolds – the most important exceptions being *homogeneous spaces* and *principal fiber bundles* (often referred to as group or Klein models and their generalization à la Cartan [21]). Hence, it should not be a big surprise that there is an intimate relationship between engineering black-box models and these two types of global models for a smooth manifold $\mathbf{M}$ (cf. Fig.1) .

Homogeneous spaces: Let a Lie group $\mathbf{G}$ act from the left on a manifold $\mathbf{M}$ in a *transitive* manner, i.e., $\mathbf{G}$ as a manifold is at least as "flexible" as $\mathbf{M}$. Take any point $Z \in \mathbf{M}$ and its stabilizer subgroup $\mathbf{H} \subset \mathbf{G}$ (acting on $\mathbf{G}$ from the right). Removing fixed points from the $\mathbf{G}$-action results in the projection $\pi : \mathbf{G} \to \mathbf{G}/\mathbf{H}$ on the quotient space $\mathbf{G}/\mathbf{H}$ (the homogeneous space or orbit space of right $\mathbf{H}$-cosets of $\mathbf{G}$). This is the classical case of a group or Klein model for a smooth manifold.

Principal fiber bundles: Group models also have a more general and enlightening fiber bundle interpretation. The above projection $\pi : \mathbf{G} \to \mathbf{G}/\mathbf{H}$ defines a smooth principal right $\mathbf{H}$-bundle $\mathbf{H} \to \mathbf{G} \to \mathbf{G}/\mathbf{H}$ with fiber or structure group $\mathbf{H}$, total space $\mathbf{G}$ and basis $\mathbf{G}/\mathbf{H}$. Moreover, assuming $\mathbf{G}$ to be any smooth manifold, the theory of principal fiber bundles ensures a unique smooth manifold structure for the quotient space $\mathbf{G}/\mathbf{H}$ precisely when the right $\mathbf{H}$-action $\mathbf{G} \times \mathbf{H} \to \mathbf{G}$ is *free* (of fixed points) and proper. Since $\mathbf{G}$ needs not to be a Lie group, this theory also applies when the port numbers of M and Z in Fig.1 differ.

In view of modeling, the main point is that in both cases $\mathbf{G}/\mathbf{H} \cong \mathbf{M}$ is a diffeomorphism. In fact, for the number of continuous parameters we have

$$\dim \mathbf{M} = \dim \mathbf{G}/\mathbf{H} = \dim \mathbf{G} - \dim \mathbf{H}. \tag{3}$$

In classical network synthesis the most familiar and elementary examples of such models are obtained by assigning dissipation and storage of energy in a mutually exclusive manner to the black boxes $\mathbf{G}$ and Z in Fig.1.

Darlington models (Resistance extraction): Darlington's theorem (cf. [18]) ensures that any strictly passive n-port admits a realization like the cascade connection in Fig. 1, where $\mathbf{G}$ is a lossless $2n$-port, i.e $\mathbf{G} \cong \mathbf{O}_{n,n}$ and Z consists of n decoupled resistors, i.e. its scattering matrix equals $S = 0_n$. Clearly, this is only a "spot-frequency" realization (for a full appreciation of classical passive Darlington synthesis of lumped element dynamical n-ports it is necessary to deal with rational matrices from $\mathrm{rat}\mathbf{O}_{n,n}$ or, more precisely, with the physical semigroup that is generated by elementary sections with positive values of the reactances). Since the subgroup of lossless $2n$-ports that leaves the subspace $\mathrm{graph}(0_n)$ invariant is easily shown to be $\mathbf{H} \cong \mathbf{O}_n \times \mathbf{O}_n$, we have a model $\mathbf{M} \cong \mathbf{O}_{n,n}/(\mathbf{O}_n \times \mathbf{O}_n)$ for passive resistive n-ports or (equivalently) the set of strictly contractive matrices. This is only an open and dense subset of the Grassmannian $\mathbf{Gr}_n \cong \mathbf{O}_{2n}/(\mathbf{O}_n \times \mathbf{O}_n)$. Other $\mathbf{O}_{n,n}$-orbits correspond to non-passive generalizations of Darlington's theorem (cf. [1], [17]) and may be labeled by the triple (n_+, n_-, n_0), $n_+ + n_- + n_0 = n$, of the numbers of positive, negative and zero resistances in Z [8]. The pertaining models $\mathbf{O}_{n,n}/\mathbf{H}$ differ only by the specific stabilizers $\mathbf{H}$. Note that $\dim \mathbf{H}$ increases with n_0; hence, there will be a loss of parameters in $\mathbf{M} \cong \mathbf{G}/\mathbf{H}$ via eq.(3). This is a rather important point in the design of identification algorithms (see e.g. [7]).

State space models (Reactance extraction): As is well-known [5], the space $\mathbf{M}(s)$ of proper rational transfer functions $M(s)$ of fixed McMillan degree n may be conceived as the orbit space $\mathbf{\Sigma}_{n,m}$ of minimal systems $\widetilde{\mathbf{\Sigma}}_{n,m} := \{A, B, C, D\}$ modulo the $\mathbf{GL}_n$ action

$$\varphi : \mathbf{GL}_n \times \widetilde{\mathbf{\Sigma}}_{n,m} \to \widetilde{\mathbf{\Sigma}}_{n,m}, \quad (T, A, B, C, D) \mapsto (TAT^{-1}, TB, CT^{-1}, D), \tag{4}$$

i.e. $\mathbf{M}(s) \cong \mathbf{\Sigma}_{n,m} := \widetilde{\mathbf{\Sigma}}_{n,m}/\mathbf{GL}_n$, where $\mathbf{\Sigma}_{n,m}$ is an analytic manifold and the canonical projection $\pi : \widetilde{\mathbf{\Sigma}}_{n,m} \to \mathbf{\Sigma}_{n,m}$ is a principal $\mathbf{GL}_n$ bundle. All this translates 1:1 into Youla's reactance extraction approach [24] to state space systems by invoking the principal fiber bundle interpretation of Fig. 1. Identifying $\mathbf{G} \cong \widetilde{\mathbf{\Sigma}}_{n,m}$ and $Z = \mathrm{graph}(s\mathbf{1}_n)$ (the external behavior of n unit-normalized reactances), the structure group of the bundle $\mathbf{H} \cong \mathbf{GL}_n$ is defined as the stabilizer of Z via its left action $\varphi_{\mathrm{L}} : \mathbf{H} \times \mathbf{Gr}_n \to \mathbf{Gr}_n$, $H\mathrm{graph}(s\mathbf{1}_n) \mapsto \mathrm{graph}(s\mathbf{1}_n)$. More explicitly:

$$\mathbf{H}\ \mathrm{span}\begin{bmatrix} s\mathbf{1}_n \\ \mathbf{1}_n \end{bmatrix} = \mathrm{span}\begin{bmatrix} s\mathbf{1}_n \\ \mathbf{1}_n \end{bmatrix} \iff \begin{bmatrix} T & \\ & T \end{bmatrix}\begin{bmatrix} s\mathbf{1}_n \\ \mathbf{1}_n \end{bmatrix} = \begin{bmatrix} s\mathbf{1}_n \\ \mathbf{1}_n \end{bmatrix} T, \qquad T \in \mathbf{GL}_n.$$

This way, internal state space transformations $T \in \mathbf{GL}_n$ gain a strikingly geometric and natural interpretation via the stabilizer of a certain space in a cascade model. Moreover, invariance of a transfer function $M(s)$ under internal equivalence transformations by some $H \in \mathbf{H}$ is clear from invariance of Z under left actions of $\mathbf{H} = \mathrm{stab}(Z)$ in Fig. 1. This may also serve as a motivating and plausible *geometric* alternative to the customary brute force algebraic proof of invariance of the transfer function.

4 Concluding Remarks

When circuit engineers state that the interconnection of passive devices results in a passive device, or when they characterize an n-ports implicitely by an atlas of charts instead of pinning it down to a fixed matrix representation, they think intuitively in a coordinate-free way. It becomes more and more evident that the nature of the rationale behind this engineering black-box thinking is a geometric-topological one. Since the mid-seventies, mathematical systems theory emphasizing the role of Lie group theory, differential geometry, and global analysis, went the other way: One started from matrix equations and reconstructed coordinate-free objects by replacing matrices with a systematic use of abstract linear vector spaces while looking for appropriate compactifications. However, there are still quite a few examples in linear systems theory where the identification of the underlying *geometric* structure is obstructed by the coordinates implicit in the customary algebraic definition of objects. Hence, there is to presume a tremendous potentiality in exploiting the crossroads of modern mathematical systems theory (geometry, topology, global analysis) and engineering multiport theory (the latter being definitely more oriented to physical systems).

References

[1] J. A. Ball and J. W. Helton. Lie groups over the field of rational functions, signed spectral factorization, signed interpolation, and amplifier design. *J. Operator Theory*, 8:19–64, 1982.

[2] R. W. Brockett. Some geometrical questions in the theory of linear systems. *IEEE Trans. Automatic Control*, AC-21:449–455, 1976.

[3] R. W. Brockett. Control theory and analytical mechanics. In C. Martin and R. Hermann, editors, *The 1976 AMES Research Center (NASA) Conference on Geometric Control Theory*, pages 1–48. Math Sci Press, Brookline. MA, 1977.

[4] C. I. Byrnes and T. E. Duncan. On certain topological invariants arising in system theory. In *New Directions in Applied Mathematics*, pages 29–71. Springer, 1982.

[5] C. I. Byrnes and C. Martin, editors. *Geometrical Methods for the Theory of Linear Systems*. Nato Advanced Study Institute, Harvard University, Reidel, 1979.

[6] W. Cauer. Ideale Transformatoren und lineare Transformationen. *Elektrische Nachrichtentechnik*, 1932.

[7] K. Diepold and R. Pauli. A Schur-type algorithm for triangular factorization of positive semidefinite matrices. In Ed F. Deprettere and Alle-Jan van der Veen, editors, *Algorithms and VLSI Architectures, Vol. B*, pages 33–42. Elsevier, Amsterdam, 1991. ISBN 0 444 89120 X.

[8] K. Diepold and R. Pauli. A recursive algorithm for lossless embedding of non-passive systems. In P. Dewilde, M.A. Kaashoek, and M. Verhaegen, editors, *Challenges of a Generalized System Theory*, pages 209–222, Amsterdam, 1993. Koninklijke Nederlandse Akademie van Wetenschappen, North-Holland.

[9] K. Diepold and R. Pauli. Actions of noncompact groups and algorithm design: A case study. In *Proc. ICASSP'97*, pages I–47 – I–50, München, 1997.

[10] J. Dieudonné. *Grundzüge der modernen Analysis, III*. Vieweg, Braunschweig, 1976.

[11] B. F. Doolin and C. F. Martin. *Introduction to Differential Geometry for Engineers*. Number 136 in Pure and Applied Mathematics. Marcel Dekker, New York, 1990.

[12] Uwe Helmke and J. B. Moore. *Optimization and Dynamical Systems*. Springer, London, 1994.

[13] J. W. Helton. Non-euclidean functional analysis and electronics. *Bulletin of the Amer. Math. Soc.*, 7:1–64, 1982.

[14] J. W. Helton. *Operator Theory, Analytic Functions, Matrices, and Electrical Engineering*, volume 68 of *CBMS Reg. Conf. Series*. Amer. Math. Soc., Providence, Rhode Island, 1987. (= Expository Lectures from the CBMS Reginal Conference held at Lincoln, Nebraska, Aug. 1985).

[15] R. E. Kalman. Mathematical description of linear dynamical systems. *SIAM J. Control*, A1:152–192, 1963.

[16] A. T. Lundell and S. Weingram. *The Topology of CW Complexes*. Van Nostrand Reinhold, 1969.

[17] W. Mathis and R. Pauli. Network theorems. In J. G. Webster, editor, *Wiley Encyclopedia of Electrical and Electronics Engineering*, volume 14, pages 227–240. Wiley, 1999.

[18] R. W. Newcomb. *Linear Multiport Synthesis*. Reinhold, New York, 1966.

[19] R. Pauli. The algebra of 2n-port transformations. In C.I. Byrnes and C.F. Martin, editors, *Linear Circuits, Systems and Signal Processing*, pages 81–86. North Holland, Amsterdam, 1988.

[20] S. H. Sattinger and S. L. Weaver. *Lie Groups and Algebras with Applications to Physics*. Springer, 1986.

[21] R. W. Sharpe. *Differential Geometry: Cartan's Generalization of Klein's Erlangen Program*. Springer, New York, 1997.

[22] A. Weissfloch. *Schaltungstheorie und Meßtechnik des Dezimeter- und Zentimeterwellengebietes*. Birkhäuser, Basel, 1954.

[23] D. C. Youla. Formulation of the "spot" frequency theory of linear time-invariant networks. Memo 99 PIBMRI-1229-64, Polytechnic Institute of Brooklyn, June 1964.

[24] D. C. Youla and P. Tissi. n-port synthesis via reactance extraction – Part I. *IEEE International Convention Record*, Part 7:183–208, 1966.

BOND GRAPHS AND MATROIDS

A. Reibiger[1] **and H. Loose**[2]

[1] TU Dresden, Fakultät Elektrotechnik, Professur für Theoretische Elektrotechnik,
Mommsenstr. 13, D-01062 Dresden, Germany
reibiger@iee.et.tu-dresden.de

[2] TU München, Lehrstuhl für Netzwerktheorie und Signalverarbeitung,
Arcisstr. 21, D-80290 München, Germany
Hannes.Loose@nws.e-technik.tu-muenchen.de

Abstract: *We define bond graphs in a completely new way, and thereby justify the bond graph terminology in the framework of classical network theory. We discuss some interrelations between bond graphs and Minty networks.*

1 Introduction

The description of the topological structure of a network by means of an oriented graph goes back to KIRCHHOFF's classical paper in 1847. Nowadays, the network theory is well established not only in electrical engineering, but almost everywhere. All the networks defined on the base of oriented graphs share the disadvantage that they are dualizable only if their graphs are planar.

MINTY [7] introduced in 1966 the notion of the graphoid, which is highly appropriate for the treatment of matroids as generalizations of graphs. Applying oriented graphoids, he extended the concept of monotone resistive networks with uncoupled branches so that those generalized resistive networks are always dualizable. It is easy to transfer this new concept to networks with more general voltage-current relations. Because of the shortcoming, that no straightforward algorithm has been found yet, which delivers networks of this kind as models for sufficient complicated technical or physical systems, this class of networks has not found considerable attention in technical literature. However, the solution sets of such networks can always be generated by means of bond graphs [13, 11, 5]. Furthermore, by giving a new definition of bond graphs we circumvent some difficulties with bond graph terminology described e.g. in [9, 2].

The application of electrical networks as models for non-electrical systems has always been accompanied by discussions about the "correct" analogy to be used, e.g. whether models for mechanical systems should base on the ordered pairs of correspondences (*voltage*$\triangleq$*velocity*, *current*$\triangleq$*force*) or (*voltage*$\triangleq$*force*, *current*$\triangleq$*velocity*). Clearly, if one uses network models which are always dualizable, like the networks based on the ideas of MINTY, these discussions are insignificant.

2 Kirchhoff and Minty Networks

In this section we define Kirchhoff and Minty networks as ordered pairs of a skeleton and a voltage-current relation. The skeleton describes the topological structure of these networks, and the voltage-current relation is a binary relation that includes all admissible pairs of voltages and currents of such a network.

The skeleton of a Kirchhoff network describes the branch-node structure of the network by means of an ordered pair of oriented graphs, and determines in that manner the meshes and cutsets of such a network implicitly. The skeleton of a Minty network is defined as an ordered pair of graphoids. Graphoids are essentially defined by the sets of their meshes and cut sets without using nodes (a minimum of necessary details about graphoids we have collected in an appendix at the end of this paper).

An oriented graph is defined as an ordered triple $(\mathcal{Z}, \mathcal{K}, \mathcal{A})$ of two disjoint sets $\mathcal{Z}$ and $\mathcal{K}$ with $(\mathcal{K} = \emptyset \Rightarrow \mathcal{Z} = \emptyset)$ and a map $\mathcal{A} : \mathcal{Z} \to \mathcal{K} \times \mathcal{K}$. $\mathcal{Z}$ is the branch set, $\mathcal{K}$ the node set, and $\mathcal{A}$ the incidence mapping of this graph. The incidence map assigns to each branch b the ordered pair $A(b) =: (v, w)$ of start and final point of b [3]. The oriented meshes and cut sets of an oriented graph are denoted by means of ordered pairs such as $(\mathcal{Z}^+, \mathcal{Z}^-)$ where $\mathcal{Z}^+, \mathcal{Z}^- \subseteq \mathcal{Z}$ are disjoint sets of branches which belong to the corresponding oriented mesh or cutset in positive or negative orientation, resp.

2.1 Agreement $\mathcal{U}$, $\mathcal{I}$, $\mathcal{P}$ and $\mathcal{T}$ denote one-dimensional oriented normed real linear spaces provided with their standard topologies. We refer to $\mathcal{U}$ as the *branch-voltage space* and to $\mathcal{I}$ as the *branch-current space*. $\mathcal{T}$ is the *time axis* and Int$\mathcal{T}$ denotes the set of all the intervals on the time axis.

$\mathcal{B} : \mathcal{U} \times \mathcal{I} \to \mathcal{P}$ denotes a non-degenerate bilinear mapping that assigns to each element $(U, I) \in \mathcal{U} \times \mathcal{I}$ the element $\mathcal{B}(U, I) \in \mathcal{P}$.

For each finite set $\mathcal{Z}$ and each interval $T \in \mathrm{Int}\mathcal{T}$ we assume that $\mathcal{U}^{\mathcal{Z}}, (\mathcal{U}^{\mathcal{Z}})^T, \ldots, (\mathcal{U}^{\mathcal{Z}})^T \times (\mathcal{I}^{\mathcal{Z}})^T$ are provided with the structure of normed linear spaces induced by that one of $\mathcal{U}$ and $\mathcal{I}$. For each $b \in \mathcal{Z}$, $\mathcal{Z}' \subset \mathcal{Z}$ ($\mathcal{Z} \neq \emptyset$), $U \in \mathcal{U}^{\mathcal{Z}}$, and $I \in \mathcal{I}^{\mathcal{Z}}$ we define by means of $\mathrm{pv}_b(U) := U(b)$, $\mathrm{pc}_b(I) := I(b)$, $\mathrm{pv}_{\mathcal{Z}'}(U) := U|\mathcal{Z}'$, and $\mathrm{pv}_{\mathcal{Z}'}(I) := I|\mathcal{Z}'$ the projections $\mathrm{pv}_b : \mathcal{U}^{\mathcal{Z}} \to \mathcal{U}$, $\mathrm{pc}_b : \mathcal{I}^{\mathcal{Z}} \to \mathcal{I}$, $\mathrm{pv}_{\mathcal{Z}'} : \mathcal{U}^{\mathcal{Z}} \to \mathcal{U}^{\mathcal{Z}'}$, and $\mathrm{pv}_{\mathcal{Z}'} : \mathcal{I}^{\mathcal{Z}} \to \mathcal{I}^{\mathcal{Z}'}$, resp. □

As usual, for any two sets X and Y we denote by Y^X the set of all mappings from X to Y. For each $f \in Y^X$ and $X' \subset X$ we denote with $f|X'$ the restriction of f to X'.

For *electrical networks*, the spaces introduced above are defined by $\mathcal{U} := \mathbb{RV}$, $\mathcal{I} := \mathbb{RA}$, $\mathcal{P} := \mathbb{RW}$, and $\mathcal{T} := \mathbb{Rs}$, and the bilinear mapping $\mathcal{B}$ assigns to each value $(U, I) \in \mathcal{U} \times \mathcal{I}$ the product of the scalar physical quantities U and I. For *normalized networks*, these spaces are $\mathcal{U} := \mathcal{I} := \mathcal{T} := \mathcal{P} := \mathbb{R}$, and $\mathcal{B}$ is defined by $\mathcal{B}(U, I) := UI$. In the case of the electrical networks the elements of the space $\mathcal{P}$ have the physical dimension of power. But there exist also technical relevant examples of networks, where the elements of $\mathcal{P}$ have a physical dimension different from that one of power. For examples see below and [12].

2.2 Definition Let $\mathcal{Z}$ be a finite set and $\mathcal{S}$ a set.

The set $\mathcal{S}$ is the *universal signal set* on $\mathcal{Z}$ if it satisfies the condition $\mathcal{S} = \bigcup_{T \in \mathrm{Int}\mathcal{T}} (\mathcal{U}^{\mathcal{Z}})^T \times (\mathcal{I}^{\mathcal{Z}})^T$. □

2.3 Definition Let $\mathcal{S}$ be a universal signal set on some finite set.

The elements of $\mathcal{S}$ are called *signals*. Let (u, i) be a signal of $\mathcal{S}$. u is the *voltage* and i is the *current* of the signal (u, i). The interval $T := \mathrm{dom}u = \mathrm{dom}i$ is called the *domain of the signal* (u, i).

Let $\mathcal{W} \subseteq \mathcal{S}$. The set $\mathrm{sd}(\mathcal{W}) := \{T \mid \exists_{(u,i) \in \mathcal{W}} T = \mathrm{dom}u\}$ is the *set of domains of the signals of* $\mathcal{W}$. $\mathcal{W}$ is called *restriction compatible* if it satisfies $\forall_{(u,i) \in \mathcal{W}} \forall_{T \in \mathrm{Int}\mathcal{T}} \big(T \subseteq \mathrm{dom}u \Rightarrow (u|T, i|T) \in \mathcal{W}\big)$. □

2.4 Remark The universal signal set contains very complicated signals, e.g. signals with nowhere differentiable voltages and currents. Therefore we usually use more specific signal sets $\mathcal{S}' \subset \mathcal{S}$. We omit a formal definition of such sets here, but typical examples are signal sets including all continuous, continuously differentiable, or piecewise continuously differentiable voltages and currents of the universal signal set. □

2.5 Definition A *K-skeleton* is an ordered pair $(\mathcal{G}_\mathrm{v}, \mathcal{G}_\mathrm{c})$ of two oriented graphs $\mathcal{G}_\mathrm{v}$ and $\mathcal{G}_\mathrm{c}$ with the same branch and node set, and which differ at most with respect to their orientation.

Let $\mathcal{C} =: (\mathcal{G}_\mathrm{v}, \mathcal{G}_\mathrm{c})$ be a K-skeleton. We call $\mathcal{G}_\mathrm{v}$ the *voltage graph* and $\mathcal{G}_\mathrm{c}$ the *current graph* of $\mathcal{C}$. A finite set is the *branch set* (*node set*, resp.) of $\mathcal{C}$ if it is the branch set (node set, resp.) of the voltage graph of $\mathcal{C}$.

A *M-skeleton* is an ordered pair $(\mathcal{G}_\mathrm{v}, \mathcal{G}_\mathrm{c})$ of two oriented graphoids $\mathcal{G}_\mathrm{v}$ and $\mathcal{G}_\mathrm{c}$ with the same branch set and which differ at most with respect to their orientation.

If $\mathcal{C} =: (\mathcal{G}_\mathrm{v}, \mathcal{G}_\mathrm{c})$ is a M-skeleton, then $\mathcal{G}_\mathrm{v}$ is its *voltage graphoid* and $\mathcal{G}_\mathrm{c}$ is its *current graphoid*. A finite set is the *branch set* of $\mathcal{C}$ if it is the branch set of the voltage graphoid of $\mathcal{C}$.

An ordered pair is a *skeleton* if it is either a K-skeleton or a M-skeleton.

Let $\mathcal{C} =: (\mathcal{G}_\mathrm{v}, \mathcal{G}_\mathrm{c})$ be a skeleton with branch set $\mathcal{Z}$ and $\emptyset \subset \mathcal{Z}' \subset \mathcal{Z}$. The ordered pair $\mathcal{C}_{\mathcal{Z}'} := (\mathcal{G}_{\mathrm{v},\mathcal{Z}'}, \mathcal{G}_{\mathrm{c},\mathcal{Z}'})$ is the *subskeleton of* $\mathcal{C}$ *defined by* $\mathcal{Z}'$ if $\mathcal{G}_{\mathrm{v},\mathcal{Z}'}$ and $\mathcal{G}_{\mathrm{c},\mathcal{Z}'}$ are the subgraphs of $\mathcal{G}_\mathrm{v}$ and $\mathcal{G}_\mathrm{c}$ induced by $\mathcal{Z}'$, resp. □

2.6 Definition An ordered pair $(\mathcal{C}, \mathcal{V})$ is a *Kirchhoff network* (resp. *Minty network*) if $\mathcal{C}$ is a K-skeleton (resp. M-skeleton) and $\mathcal{V}$ is a non-void, restriction compatible subset of the universal signal set on the branch set of $\mathcal{C}$. $\mathcal{V}$ is the *voltage-current relation* of the network $(\mathcal{C}, \mathcal{V})$.

An ordered pair is a *network* if it is either a Kirchhoff network or a Minty network.

2.7 Agreement Let $\mathcal{N} =: ((\mathcal{G}_\mathrm{v}, \mathcal{G}_\mathrm{c}), \mathcal{V})$ be a network with branch set $\mathcal{Z}$. The orientations of the branches in $\mathcal{G}_\mathrm{v}$ and $\mathcal{G}_\mathrm{c}$ are the reference directions for the branch voltages and branch currents. With $\mathcal{Z}^{\mathrm{ass}\cdot}$ we denote the subset of branches with associated reference directions for the branch voltages and branch currents. □

2.8 Definition Let $\mathcal{N}$ be a network with the branch set $\mathcal{Z}$. Let $\mathcal{S}$ be its universal signal set.

For each signal $(u, i) \in \mathcal{S}$ and each branch $b \in \mathcal{Z}$ we denote with u_b and i_b the time functions defined by $u_b := \mathrm{pv}_b \circ u$ and $i_b := \mathrm{pc}_b \circ i$, resp. We call the functions u_b and i_b the *branch-voltages* and the *branch-currents* of the signal (u, i), resp. Similarly, we denote for each $(u, i) \in \mathcal{S}$ and each non-empty set $\mathcal{Z}' \subset \mathcal{Z}$ by $u_{\mathcal{Z}'}$ and $i_{\mathcal{Z}'}$ the time functions defined by $u_{\mathcal{Z}'} := \mathrm{pv}_{\mathcal{Z}'} \circ u$ and $i_{\mathcal{Z}'} := \mathrm{pc}_{\mathcal{Z}'} \circ i$, resp. The functions $u_{\mathcal{Z}'}$ and $i_{\mathcal{Z}'}$ ($\emptyset \subset \mathcal{Z}' \subset \mathcal{Z}$) are the *partial voltages* and *partial currents* of the signal (u, i).

For each $\mathcal{W} \subseteq \mathcal{S}$ and for each $\mathcal{Z}'$ with $\emptyset \subset \mathcal{Z}' \subset \mathcal{Z}$ the set $\mathcal{W}_{\mathcal{Z}'}$ is defined by $\mathcal{W}_{\mathcal{Z}'} := \{(u_{\mathcal{Z}'}, i_{\mathcal{Z}'}) \mid (u, i) \in \mathcal{W}\}$. □

2.9 Definition Let $\mathcal{N} =: ((\mathcal{G}_\mathrm{v}, \mathcal{G}_\mathrm{c}), \mathcal{V})$ be a network with universal signal set $\mathcal{S}$.

A signal $(u, i) \in \mathcal{S}$ is called a *Kirchhoff signal* if the following two conditions hold true:

(KVL) For every oriented mesh $(\mathcal{Z}^+, \mathcal{Z}^-)$ (respectively $\{\mathcal{Z}^+, \mathcal{Z}^-\}$) of $\mathcal{G}_\mathrm{v}$ the voltage u satisfies

$$\sum_{b \in \mathcal{Z}^+} u_b - \sum_{b \in \mathcal{Z}^-} u_b = 0 .$$

(KCL) For every oriented cutset $(\mathcal{Z}^+, \mathcal{Z}^-)$ (respectively $\{\mathcal{Z}^+, \mathcal{Z}^-\}$) of $\mathcal{G}_c$ the current i satisfies

$$\sum_{b\in\mathcal{Z}^+} i_b - \sum_{b\in\mathcal{Z}^-} i_b = 0.$$

The conditions (KVL) and (KCL) are called *Kirchhoff voltage law* and *Kirchhoff current law*, resp. □

2.10 Definition Let $\mathcal{N}$ be a network with universal signal set $\mathcal{S}$.
A subset $\mathcal{H} \subseteq \mathcal{S}$ is the *Kirchhoff part* of the universal signal set of $\mathcal{N}$ if it is the set of all Kirchhoff signals of $\mathcal{S}$. The intersection $\mathcal{L} := \mathcal{H} \cap \mathcal{V}$ is the *solution set* of $\mathcal{N}$. The elements of $\mathcal{L}$ are called the *solutions* of $\mathcal{N}$.

2.11 Definition Let $\mathcal{Z}'$ be a nontrivial subset of the branch set $\mathcal{Z}$ of a network $\mathcal{N} =: (\mathcal{C}, \mathcal{V})$.
The set $\mathcal{Z}'$ and its complement $\mathcal{Z}'' = \mathcal{Z}\backslash\mathcal{Z}'$ are *uncoupled* if $\mathcal{V} = \{(u,i) \in \mathcal{S} \mid (u_{\mathcal{Z}'}, i_{\mathcal{Z}'}) \in \mathcal{V}_{\mathcal{Z}'} \wedge (u_{\mathcal{Z}''}, i_{\mathcal{Z}''}) \in \mathcal{V}_{\mathcal{Z}''}\}$.
Let $\mathcal{Z}'$ be a branch set which is uncoupled from its complement and $\mathcal{C}_{\mathcal{Z}'}$ the subskeleton of $\mathcal{C}$ defined by $\mathcal{Z}' \subset \mathcal{Z}$. Then the ordered pair $(\mathcal{C}_{\mathcal{Z}'}, \mathcal{V}_{\mathcal{Z}'}) =: \mathcal{N}_{\mathcal{Z}'}$ is a network. We call it the *subnetwork* of $\mathcal{N}$ defined by $\mathcal{Z}'$.
The network $\mathcal{N}$ is an *elementary network* if its skeleton consists of single-branch components only and no nontrivial subset of its branch set defines a subnetwork of $\mathcal{N}$. □
The following definition is essential for our approach to use bond graphs for a description of Minty networks.

2.12 Definition Let $\mathcal{N}$ and $\tilde{\mathcal{N}}$ be two networks with the branch sets $\mathcal{Z}$ and $\tilde{\mathcal{Z}}$, resp., and the solution sets $\mathcal{L}$ and $\tilde{\mathcal{L}}$, resp. Let $\mathcal{Z} \subseteq \tilde{\mathcal{Z}}$ and $\tilde{\mathcal{L}}_{\mathcal{Z}} := \{(u_{\mathcal{Z}}, i_{\mathcal{Z}}) \mid (u,i) \in \tilde{\mathcal{L}}\}$.
The network $\tilde{\mathcal{N}}$ *generates the solutions of* $\mathcal{N}$ if $\tilde{\mathcal{L}}_{\mathcal{Z}} = \mathcal{L}$.

2.13 Proposition *For every Kirchhoff network there exists a Minty network with the same branch set generating its solutions.* □

The proof of the assertion above is trivial since every oriented graph induces an oriented graphoid. Obviously, the reversal of the above assertion is not true because not every Kirchhoff network has a dual one.

Let $\mathcal{G}$ be an oriented graph or an oriented graphoid. Then we denote in the following by $\mathcal{G}^*$ an oriented graph which is dual to $\mathcal{G}$ and has the same branch set as $\mathcal{G}$ or an oriented graphoid which is the dual one of $\mathcal{G}$, resp. (c.f. proposition A.5).

2.14 Definition Let $\mathcal{C} =: (\mathcal{G}_v, \mathcal{G}_c)$ be a skeleton. We call the skeleton $\mathcal{C}^* := (\mathcal{G}_c^*, \mathcal{G}_v^*)$ the *dual skeleton of* $\mathcal{C}$. □

2.15 Definition Let $\mathcal{N} =: (\mathcal{C}, \mathcal{V})$ and $\mathcal{N}^\circ =: (\mathcal{C}^\circ, \mathcal{V}^\circ)$ be two networks. $\mathcal{N}^\circ$ is a *dual network* to $\mathcal{N}$ if $\mathcal{C}^\circ = \mathcal{C}^*$ and if there exists a linear bijection $R : \mathcal{I} \to \mathcal{U}$ so that $\forall_{u,i}\ \big((u,i) \in \mathcal{V} \Leftrightarrow (R_0 \circ i, R_0^{-1} \circ u) \in \mathcal{V}^\circ\big)$, where $R_0 : \mathcal{I}^{\mathcal{Z}} \to \mathcal{U}^{\mathcal{Z}}$ is the linear bijection defined by $\forall_{b\in\mathcal{Z}}\ (R_0(I))(b) = R(I(b))$. □

2.16 Proposition *Every Minty network has a dual Minty network.* □

The essential difference between Kirchhoff and Minty networks is the fact that not for every Kirchhoff network exists a dual Kirchhoff network.

2.17 Remark On this base it is possible to introduce *resistive, inductive, capacitive networks, nullators, norators*, etc. Furthermore, we can define the representation of networks as *interconnection* of subnetworks. □

2.18 Definition Let $\mathcal{S}$ be a subset of the universal signal set on some set $\mathcal{Z}$, $(u,i) \in \mathcal{S}$, and ζ be a bijective mapping from $\mathcal{Z}$ to $\{1, 2, \ldots, |\mathcal{Z}|\}$.
We call the ordered pair $(\boldsymbol{u}, \boldsymbol{i}) := \left({}^t\left(u_{\zeta(1)}\, u_{\zeta(2)} \ldots u_{\zeta(|\mathcal{Z}|)}\right), {}^t\left(i_{\zeta(1)}\, i_{\zeta(2)} \ldots i_{\zeta(|\mathcal{Z}|)}\right)\right)$ of column matrices the *matrix representation of the signal* $(u,i) \in \mathcal{S}$ *with respect to* ζ. □

For each finite set X we denote with $|X|$ the number of elements included in X.

2.19 Definition Let $\mathcal{N} =: (\mathcal{C}, \mathcal{V})$ be a Kirchhoff network with branch set $\mathcal{Z}$ ($|\mathcal{Z}| \geq 2$) and $\mathcal{Z}^{\text{ass.}}$ the set of branches with associated reference directions for branch voltages and branch currents. $\mathcal{S}$ is a signal set included in the universal signal set on $\mathcal{Z}$. ζ is a bijective mapping $\zeta : \{1, \ldots, |\mathcal{Z}|\} \longrightarrow \mathcal{Z}$. $\boldsymbol{D}$ denotes a $|\mathcal{Z}| \times |\mathcal{Z}|$ diagonal matrix with $\boldsymbol{D}(\zeta^{-1}(b), \zeta^{-1}(b)) = +1$ if $b \in \mathcal{Z}^{\text{ass.}}$ and $\boldsymbol{D}(\zeta^{-1}(b), \zeta^{-1}(b)) = -1$ if $b \in \mathcal{Z} \setminus \mathcal{Z}^{\text{ass.}}$.
The network $\mathcal{N}$ is an *ideal transformer* if there exist two matrices $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{Q}^{|\mathcal{Z}|\times|\mathcal{Z}|}$, satisfying the two conditions $\operatorname{rank}(\boldsymbol{A}\ \boldsymbol{B}) = |\mathcal{Z}|$ and ${}^t\boldsymbol{A}\boldsymbol{B} = \boldsymbol{0}$, so that the voltage-current relation $\mathcal{V}$ is equal to the set of all signals $(u,i) \in \mathcal{S}$ whose matrix representation with respect to ζ is a solution of the system of equations $\boldsymbol{A}\boldsymbol{u} + \boldsymbol{B}\boldsymbol{D}\boldsymbol{i} = \boldsymbol{0}$. □

2.20 Remark The above definition of an ideal transformer was given by A. E. SEN and Y. TOKAD in 1980. Special cases of ideal transformers are *ideal transformer of first kind* where $\operatorname{rank}\boldsymbol{A} = |\mathcal{Z}| - 1$ and *ideal transformer of second kind* where $\operatorname{rank}\boldsymbol{B} = |\mathcal{Z}| - 1$, which were already introduced in 1932 by W. CAUER. The parallel connection multiport and the series connection multiport introduced in the next section are special cases of those two networks. Another special case are the ideal transformers which have two branches only and which therefore belong to both, the class of ideal transformers of first and of second kind. □

2.21 Definition Let $\mathcal{N}$ be a network with an even number of branches. $\mathcal{N}$ is a network in *Belevitch normal form* if there exists a partitioning of its branch set into two sets $\mathcal{Z}^{\text{tr.}}$ and $\mathcal{Z}^{\text{ext.}}$ with $|\mathcal{Z}^{\text{tr.}}| = |\mathcal{Z}^{\text{ext.}}|$ so that the

subnetwork $\mathcal{N}_{\mathcal{Z}^{tr}}$ is an ideal transformer and all components of the skeleton of $\mathcal{N}$ consist of exactly two parallel branches and one branch of each component is an element of $\mathcal{Z}^{tr}$. □

2.22 Theorem *For every network there exists a network in Belevitch normal form generating its solutions.*

For the class of Kirchhoff networks a proof is given in [1]. An extended version of this proof involving the class of Minty networks is given in [5].

3 Paynter Networks and Bond Graphs

3.1 Definition Let $\mathcal{N} =: (\mathcal{C}, \mathcal{V})$ be an elementary Kirchhoff network with voltage graph $\mathcal{G}_v$, current graph $\mathcal{G}_c$, and branch set $\mathcal{Z}$ ($|\mathcal{Z}| \geq 2$), and let $(\sigma_b)_{b\in\mathcal{Z}}$ be a family with $\sigma_b = +1$ if $b \in \mathcal{Z}^{ass.}$ and $\sigma_b = -1$ otherwise. $\mathcal{S}$ denotes a signal set included in the universal signal set on the branch set of $\mathcal{N}$.

$\mathcal{N}$ is a *series* or *parallel connection multiport* if its voltage-current relation $\mathcal{V}$ is the set of all signals $(u, i) \in \mathcal{S}$ satisfying the condition $\left(\sum_{b\in\mathcal{Z}} \sigma_b u_b = 0 \wedge \forall_{a,b\in\mathcal{Z}}\, i_a = i_b\right)$ or $\left(\sum_{b\in\mathcal{Z}} \sigma_b i_b = 0 \wedge \forall_{a,b\in\mathcal{Z}}\, u_a = u_b\right)$, resp. □

3.2 Definition A Kirchhoff network $\mathcal{N}$ is called a *Paynter network* if it fulfills the following conditions:

(P1) The components of the skeleton of $\mathcal{N}$ contain exactly two parallel branches and two nodes. The two branches of every component are oriented in such a manner that they are a similarly oriented cutset of the voltage graph of $\mathcal{N}$ and a similarly oriented mesh of the current graph of $\mathcal{N}$.

(P2) There exist three disjoint sets $\mathfrak{N}^{ext.}$, $\mathfrak{N}^{\nabla}$, and $\mathfrak{N}^{O}$ of elementary networks with $\mathfrak{N}^{ext.} \neq \emptyset$ and $\mathfrak{N}^{\nabla} \cup \mathfrak{N}^{O} \neq \emptyset$ where the networks in $\mathfrak{N}^{\nabla}$ and $\mathfrak{N}^{O}$ are series and parallel connection multiports, resp., so that $\mathcal{N}$ is an interconnection of all the networks of $\mathfrak{N} := \mathfrak{N}^{ext.} \cup \mathfrak{N}^{\nabla} \cup \mathfrak{N}^{O}$.

(P3) The two branches of every component of the skeleton of $\mathcal{N}$ belong to two different networks of $\mathfrak{N}$, and at least one branch belongs to a network of $\mathfrak{N}^{\nabla} \cup \mathfrak{N}^{O}$.

The branches of the networks of $\mathfrak{N}^{ext.}$ are the *external branches* of $\mathcal{N}$. The branches of the networks of $\mathfrak{N}^{\nabla} \cup \mathfrak{N}^{O}$ are the *internal branches* of $\mathcal{N}$. The subnetwork of $\mathcal{N}$ which is the interconnection of all the networks of $\mathfrak{N}^{\nabla} \cup \mathfrak{N}^{O}$ is the *interconnection network* of $\mathcal{N}$. □

3.3 Remark It is obvious that each Paynter network is a network in Belevitch normal form. □

3.4 Definition An oriented graph $(\mathfrak{B}, \mathfrak{N}, \mathfrak{H})$ is a *bond graph* if it satisfies the following conditions:

(B1) There exist a Paynter network $\mathcal{N}$ and three sets $\mathfrak{N}^{ext.}$, $\mathfrak{N}^{\nabla}$, $\mathfrak{N}^{O}$ of elementary subnetworks of this network satisfying the conditions given in definition 3.2.

(B2) The elements of $\mathfrak{B}$ are the ordered pairs of the start and final points of the branches of the components of the voltage graph of $\mathcal{N}$.

(B3) $\mathfrak{N} = \mathfrak{N}^{ext.} \cup \mathfrak{N}^{\nabla} \cup \mathfrak{N}^{O}$.

(B4) $\mathfrak{H} : \mathfrak{B} \to \mathfrak{N} \times \mathfrak{N}$ is a mapping which assigns to each ordered pair $(v, w) \in \mathfrak{B}$ that pair $(\mathcal{M}, \mathcal{Q})$ of networks $\mathcal{M}$ and $\mathcal{Q}$ whose branches are incident with v and w in the skeleton of the network $\mathcal{N}$ where the branch that belongs to the (second) network $\mathcal{Q}$ is similarly directed in the voltage graph and the current graph of $\mathcal{N}$.

The elements of $\mathfrak{B}$ are the *bonds* of the bond graph. □

3.5 Remark Bond graphs in the sense of definition 3.4 can be represented by some pictorial diagrams, which are traditionally denoted as bond graphs. These diagrams are determined by means of an oriented geometric graph of $\mathbb{R}^3$ [3] which is isomorphic to the given bond graph, and a projection of this geometric graph into $\mathbb{R}^2$ which is identified with the "drawing plane", so that the restriction of this projection on the node set of the geometric graph of $\mathbb{R}^3$ is a bijective mapping.

In such diagrams, the nodes representing series connection multiports are denoted by means of the symbol "1" or "∇" and the nodes representing parallel connection multiports are denoted by the symbol "0" or "O". (That is the reason why the terms *1-junction* and *0-junction* are commonly used to denote the corresponding kinds of connection multiports.) The external multiports of a bond graph are represented by some indexed literals. The use of these literals is more or less standardized, e.g. R_j denotes the jth resistive multiport, C_k denotes the kth capacitive multiport, etc. The oriented branches of bond graphs are depicted as half arrows.

The orientations of the half arrows deliver reference directions for the power exchange between the ports if the elements of the space $\mathcal{P}$ introduced in agreement 2.1 have the physical dimension of power. Modelling the energy exchange between the parts of a technical or physical system is the usual method to obtain a bond graph as a model for such a system (c.f. [4]). Bond graphs corresponding to networks with a space $\mathcal{P}$ whose elements do not have the physical dimension of power, or are not normalized representations of such quantities, are usually called *pseudo bond graphs*.

From remark 2.20 it follows that in some cases there exist several partitions of the interconnection network into series and parallel connection multiports. Therefore such a partition needs to be prescribed for each Paynter

network in order to get a one-to-one correspondence between the class of Paynter networks and that one of bond graphs. □

3.6 Theorem *For each Minty network $\mathcal{N}$ there exists a Paynter network generating the solutions of $\mathcal{N}$.*

A first complete proof of this theorem has been given in [5]. This proof applies matrix representations of graphoids which will not be introduced here. In short, on the base of an arbitrary fundamental mesh or cutset matrix of the skeleton of the Minty network is is easy to construct a Paynter network generating its solutions. Of course, there are many different possibilities for such a construction, first, because of the chosen tree in the skeleton of the Minty network and, second, because of some bond graph equivalences that can be used to modify the constructed Paynter network without altering its solution set.

The next proposition follows immediately from the theorem above because each Kirchhoff network defines in a unique manner a Minty network with the same branch set generating the solution set of the given Kirchhoff network. The assertion of this corollary, however, is a well known theorem in bond graph literature [8, 4].

3.7 Corollary *For each Kirchhoff network $\mathcal{N}$ there exists a Paynter network $\tilde{\mathcal{N}}$ generating the solutions of $\mathcal{N}$.* □

3.8 Definition A Kirchhoff network $\mathcal{N}$ is a *Paynter multiport* if it satisfies the following conditions:

(M1) Each component of the skeleton of $\mathcal{N}$ consists either of exactly one branch or of exactly two parallel branches.

(M2) If $\mathcal{N}$ is augmented by a set of norators with appropriately oriented voltage and current graphs so that the branches of these norators are connected parallel to the branches of the one-branch components of $\mathcal{N}$, the network $\mathcal{N}^{\mathrm{aug}\cdot}$ constructed in this manner is a Paynter network. □

3.9 Remark Let $\mathcal{N}$ be a Paynter multiport and $\mathcal{N}^{\mathrm{aug}\cdot}$ the network constructed from $\mathcal{N}$ by the augmentation with some norators as described in condition (M2). Let additionally $\mathcal{Z}^{\mathrm{nor}\cdot}$ denote the branch set of these norators, $\mathcal{L}^{\mathrm{aug}\cdot}$ the set of all solutions of $\mathcal{N}^{\mathrm{aug}\cdot}$, and let $\mathfrak{G}^{\mathrm{aug}\cdot}$ be the bond graph corresponding to $\mathcal{N}^{\mathrm{aug}\cdot}$ in the sense of definition 3.4.

Then $\mathcal{L}^{\mathrm{aug}\cdot}_{\mathcal{Z}^{\mathrm{nor}\cdot}}$ delivers a description of the terminal behavior of the Paynter multiport $\mathcal{N}$ (c.f. [10]).

Let $\mathfrak{D}^{\mathrm{aug}\cdot}$ be the diagram associated to the bond graph $\mathfrak{G}^{\mathrm{aug}\cdot}$ by the construction described in remark 3.5. Deletion of the nodes which represent the added norators in $\mathfrak{D}^{\mathrm{aug}\cdot}$ leads to a new diagram. This is traditionally called the bond graph of the Paynter multiport $\mathcal{N}$. The half arrows, which were originally incident with the nodes representing the added norators in $\mathfrak{D}^{\mathrm{aug}\cdot}$, represent now the ports of $\mathcal{N}$. □

3.10 Remark The voltage-current relation of a canonical representative [10] of the terminal behaviour of the interconnection network of a Paynter network is always an ideal transformer [1]. As the examples given in [9] clearly show, the matrices of a hybrid representation of the voltage-current relation of this ideal transformer cannot generally be interpreted as fundamental mesh and cut set matrices of oriented graphs differing at most with respect to their orientations, and, moreover, the difficulties that arise in some bond graphs with "odd loops" in their graphical representation as discussed in [9] are caused by such matrices which are not even unimodular [14], i.e. by matrices which do not induce oriented graphoids! □

3.11 Remark Let $\mathcal{N}$ be a Paynter network and $\mathfrak{G} =: (\mathfrak{B}, \mathfrak{N}, \mathfrak{H})$ a corresponding bond graph. A complete forest in the skeleton of $\mathcal{N}$ can be represented as an additional orientation of the branches of $\mathfrak{G}$. By use of the notations from definition 3.4 we introduce an additional incidence mapping $\mathfrak{C} : \mathfrak{B} \to \mathfrak{N} \times \mathfrak{N}$. This mapping assigns to each ordered pair $(v, w) \in \mathfrak{B}$ the pair $(\mathcal{M}, \mathcal{Q})$ of the two networks $\mathcal{M}$ and $\mathcal{Q}$ whose branches are incident with v and w in the skeleton of $\mathcal{N}$ where the forest branch of these two branches belongs to $\mathcal{Q}$.

Let $\mathfrak{D}$ be the graphical representation of $\mathfrak{G}$ (c.f. 3.5). The orientation of the branches of the oriented graph $(\mathfrak{B}, \mathfrak{N}, \mathfrak{C})$ are traditionally represented in $\mathfrak{D}$ by means of the so called causal strokes.

It is well known that the use of appropriate forests simplifies the set up of state space equations. Independent of this, the selection of an arbitrary forest allows us to reduce the number of variables in the equations for the analysis of Paynter networks by half if a system of equations with twig voltages and link currents is used instead of a tableau system with all branch voltages and branch currents. The variables denoting the twig voltages and link currents are usually denoted as the conjugated variables associated to the corresponding bond. □

3.12 Proposition *For every Paynter network there exists a dual network, which is an Paynter network, too.* □

The skeleton of a Paynter network is always planar and therefore dualizable.

3.13 Theorem *Let $\mathcal{N}$ be a Paynter network, $\mathfrak{G} =: (\mathfrak{B}, \mathfrak{N}, \mathfrak{H})$ a bond graph corresponding to $\mathcal{N}$ in the sense of definition 3.4, and $\mathcal{N}^*$ a Paynter network dual to $\mathcal{N}$ with the same branch set as $\mathcal{N}$. Then a bond graph corresponding to $\mathcal{N}^*$ can be constructed from that one corresponding to $\mathcal{N}$ if each elementary network of $\mathfrak{N}$ is replaced by an elementary multiport with dualized voltage-current relation.*

Obviously, thereby each series connection multiport has to be replaced by a parallel connection multiport, and vice versa. The fact that each Paynter network is dualizable explains also why there is no contradiction between the

use of bond graph models for mechanical systems described in [15] based on the first pair of the correspondences, which were mentioned in the introduction, and the bond graph models described in [4] for the same purposes based on the second of these correspondences.

4 Concluding Remarks

All results presented above can be carried over on the theory of multidimensional networks developed in [12].

A Appendix

Here are given the definitions of graphoids and oriented graphoids only. For more details c.f. [7, 14, 13].

A.1 Definition and Agreement A *painting* of a finite set $\mathcal{Z}$ is a partitioning of $\mathcal{Z}$ into three disjoint subsets $\mathcal{R}$, $\mathcal{G}$, $\mathcal{B}$ so that $\mathcal{G}$ is a one-element set and $\mathcal{Z} = \mathcal{R} \cup \mathcal{G} \cup \mathcal{B}$. We regard the elements in $\mathcal{R}$ as being "painted red", the element in $\mathcal{G}$ as being "painted green", and the elements in $\mathcal{B}$ as being "painted blue". □

A.2 Definition A *graphoid* is a triple $(\mathcal{Z}, \mathfrak{M}, \mathfrak{C})$ of a finite set $\mathcal{Z}$ and two collections $\mathfrak{M}$ and $\mathfrak{C}$ of nonempty subsets of $\mathcal{Z}$, called *meshes* and *cutsets*, satisfying the conditions

(G1) $\forall_{M \in \mathfrak{M}} \forall_{C \in \mathfrak{C}} |M \cap C| \neq 1$.

(G2) For any painting of $\mathcal{Z}$ there is either a member of $\mathfrak{M}$ consisting of the green element and no blue elements or a member of $\mathfrak{C}$ consisting of the green element and no red elements.

(G3) No member of $\mathfrak{M}$ contains another member of $\mathfrak{M}$ properly; no member of $\mathfrak{C}$ contains another member of $\mathfrak{C}$ properly. □

A.3 Remark $(\mathcal{Z}, \mathfrak{M}, \mathfrak{C})$ is a graphoid if and only if the ordered pairs $(\mathcal{Z}, \mathfrak{M})$ and $(\mathcal{Z}, \mathfrak{C})$ are matroids which are dual to each other. Therefore there exists to each graphoid $(\mathcal{Z}, \mathfrak{M}, \mathfrak{C})$ a *dual* one $(\mathcal{Z}, \mathfrak{C}, \mathfrak{M})$, which is, moreover, uniquely determined. □

For a simple example consider an arbitrary non-oriented graph. The triple formed by its branch set, the set of all its meshes, and the set of all its cutsets is always a graphoid. But note that not every graph can be dualized and not every graphoid can be induced by an appropriate graph.

A.4 Definition An ordered triple $(\mathcal{Z}, \mathfrak{M}, \mathfrak{C})$ is an *oriented graphoid* if the following conditions are satisfied:

(O1) $\mathcal{Z}$ is a finite set. The elements of $\mathfrak{M}$ and $\mathfrak{C}$ are sets $\{\mathcal{Z}^+, \mathcal{Z}^-\}$ with $\mathcal{Z}^+, \mathcal{Z}^- \subseteq \mathcal{Z}$ and $\mathcal{Z}^+ \cap \mathcal{Z}^- = \emptyset$.

(O2) The triple $(\mathcal{Z}, \mathfrak{M}, \mathfrak{C})$ with $\mathfrak{M} := \{\mathcal{Z}^+ \cup \mathcal{Z}^- \mid \{\mathcal{Z}^+, \mathcal{Z}^-\} \in \mathfrak{M}\}$ and $\mathfrak{C} := \{\mathcal{Z}^+ \cup \mathcal{Z}^- \mid \{\mathcal{Z}^+, \mathcal{Z}^-\} \in \mathfrak{C}\}$ is an graphoid.

(O3) Orthogonality: $\forall_{\{\bar{\mathcal{Z}}^+, \bar{\mathcal{Z}}^-\} \in \mathfrak{M}} \forall_{\{\mathcal{Z}^+, \mathcal{Z}^-\} \in \mathfrak{C}} \left(|\bar{\mathcal{Z}}^+ \cap \mathcal{Z}^+| + |\mathcal{Z}^- \cap \bar{\mathcal{Z}}^-| = |\bar{\mathcal{Z}}^+ \cap \mathcal{Z}^-| + |\mathcal{Z}^- \cap \bar{\mathcal{Z}}^+|\right)$. □

A.5 Proposition and Definition *If $(\mathcal{Z}, \mathfrak{M}, \mathfrak{C})$ is an oriented graphoid then $(\mathcal{Z}, \mathfrak{C}, \mathfrak{M})$ is an oriented graphoid, too.* $(\mathcal{Z}, \mathfrak{C}, \mathfrak{M})$ is called the *dual graphoid* to $(\mathcal{Z}, \mathfrak{M}, \mathfrak{C})$. □

References

[1] BELEVITCH, V., *Classical Network Theory*, Holden-Day, San Francisco, 1968.

[2] BIRKETT, S. H. AND ROE, P. H., *The mathematical foundations of bond graphs – I. algrebraic theory*, J. Franklin Inst., 326 (1989), pp. 329–350.

[3] BUSACKER, R. G. AND SAATY, T. L., *Finite graphs and networks*, McGraw-Hill Book Comp., New York, 1965.

[4] KARNOPP, DEAN C., MARGOLIS, DONALD L., AND ROSENBERG, RONALD C., *System Dynamics: A Unified Approach*, John Wiley & Sons, Inc., 1990.

[5] LOOSE, H., *Beiträge zur Theorie der Bondgraphenetzwerke*. Diplomarbeit, TU Dresden, 1999.

[6] MATHIS, W. AND MARTEN, W., *On the structure of networks and duality theory*, Proc. 31th Midwest Symposium on Circuit Theory and Systems, St. Louis, Miss., USA, 1988.

[7] G. J. MINTY, *On the axiomatic foundations of the theories of directed linear graphs, electrical networks and network-programming*, J. of Math. and Mech., 15 (1966), pp. 485–520.

[8] PERELSON, A. S., *Description of electrical networks using bond graphs*, Int. J. Circuit Theory and Appl., 4 (1976), pp. 107–123.

[9] PERELSON, A. S. AND OSTER, G. F., *Bond graphs and linear graphs*, J. Franklin Inst., 302 (1976), pp. 159–185.

[10] REIBIGER, A., *Über das Klemmenverhalten von Netzwerken*, Wiss. Z. Techn. Univers. Dresden, 35 (1986), pp. 165–173.

[11] ———, *Netzwerke und Bondgraphen*, Wiss. Z. Techn. Univers. Dresden, 40 (1991), pp. 67–70.

[12] REIBIGER, A., LOOSE, H., AND NÄHRING, T., *Multidimensional networks*, Intern. Symp. on Theor. Electr. Eng. (ISTET'99), Magdeburg, Germany, 1999.

[13] REIBIGER, A. AND WITTE, K., *Graphoidale Netzwerke*, Wiss. Z. Techn. Univers. Dresden, 39 (1990), pp. 83–86.

[14] THULASIRAMAN, K. AND SWAMY, M. N. S., *Graphs: Theory and Algorithms*, John Wiley & Sons, Inc., 1992.

[15] WELLSTEAD, P. E., *Introduction to Physical System Modelling*, Academic Press Inc., 1979.

PHYSICALLY ORIENTED MODELING OF HETEROGENEOUS SYSTEMS

Peter Schwarz
Fraunhofer Institute for Integrated Circuits, Design Automation Department EAS
Zeunerstrasse 38, D-01069 Dresden; email: schwarz@eas.iis.fhg.de

Abstract. Technical systems can be characterized very often as complex heterogeneous systems. Many simulation algorithms exist for the analysis of continuous and discrete systems. However, the first modeling steps in the physical domains have not been sufficiently assisted by CAD tools. Multiports and generalized KIRCHHOFF's networks proved to be powerful concepts in a physically oriented modeling methodology. Modeling languages like VHDL-AMS and Modelica will support the practical application of these approaches.

1. Introduction

Modern technical systems like integrated circuits, micro-electro-mechanical systems (MEMS), mechatronic systems or distributed automation systems can be characterized as complex heterogeneous systems. Typically they show some of the following features:

- mixed-domain (mechanical, electrical, thermal, fluidic, ... phenomena),
- partially close coupling between these domains, side effects, cross coupling,
- distributed and lumped (concentrated) elements,
- discrete and continuous signals and systems (in electronics: analog and digital),
- very large and stiff systems of differential equations to describe the continuous subsystems.

Depending on the level of abstraction, partial differential equations (PDE) and ordinary differential equations (ODE) are the mathematical models (system equations) of continuous subsystems. We will focus on ODE (and briefly discuss the transition from PDE to ODE in sec. 3). A lot of algorithms and computer programs is available for the numerical solution of the system equations. However, there is a demand for more and better assistance in finding out these system equations. A powerful interdisciplinary **modeling methodology** is necessary to analyse real-world problems.

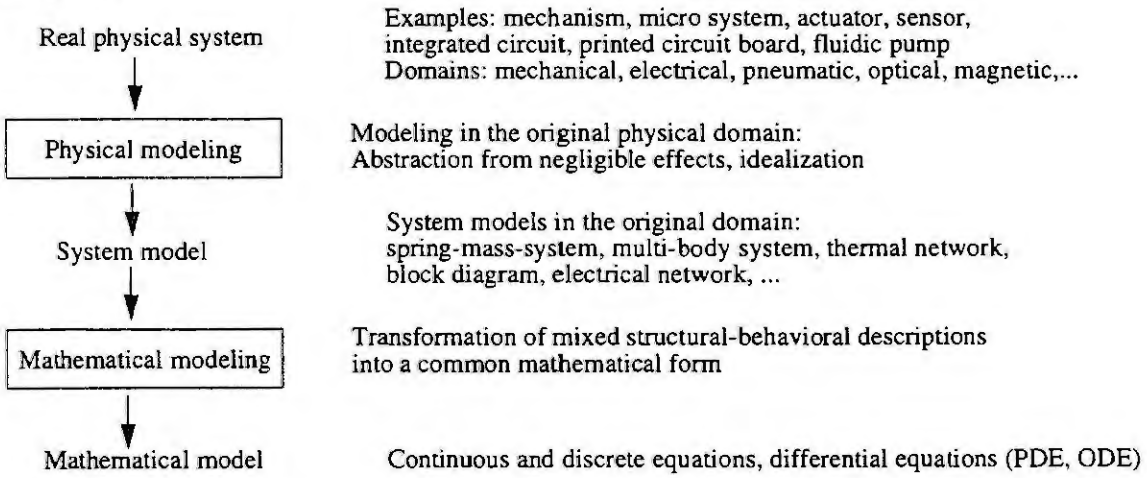


Fig. 1: Steps in modeling physical systems

Basically, the modeling process can be divided into two fundamental steps (Fig. 1). At present, the second step is mostly well supported by the input processing programs of simulators. So we will put emphasis on the first step, the "physically-oriented modeling" of complex heterogeneous systems. The term "physically-oriented" means: it is a goal of the modeling approach that as much as possible modeling steps are closely related to the design process of technical systems and to the intuitive procedure of the design engineer. Therefore, different description methods have to be considered. We can distinguish between

- behavioral description: equation-oriented, similar to the textual notation of an ODE,
- structural description: the system model is hierarchically composed of subsystems and basic elements (the so-called primitives) available with the simulators.

Mostly, system models with spatially lumped elements appear as implicit nonlinear differential-algebraic equations (DAE). Only in special cases their formulation as explicit state equations is possible.

In Fig. 2 some well-known descriptions of the same system (a simple time-continuous mechanical system with mass, spring, and damping) are shown:

- a mathematical description which may be processed by tools like Mathematica or Maple;
- the model formulated in a "hardware description language" (HDL) for the electronic system simulator Saber;
- a block diagram (or signalflow diagram) to be performed by Matlab/Simulink or MatrixX;
- a mechanical network and its equivalent electrical network (which can be simulated by electronic network analysis programs like SPICE).

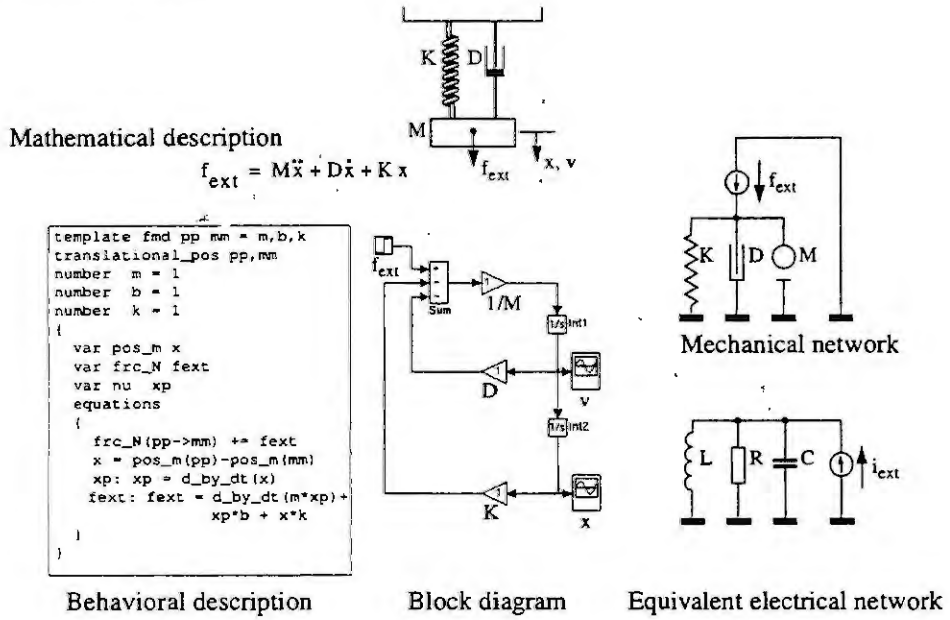


Fig. 2: Different descriptions of a mechanical system

Especially, the use of equivalent electrical networks for modeling non-electrical systems has a long and successful tradition [10], [16], [15], [18], [20], [31], [39]. However, for a long time it had been restricted to relatively small systems because of the cumbersome handling of the basic elements of older simulators like SPICE (resistance R, capacitance C, inductance L, ...) in modeling complex functionalities. Also the treatment of translational and rotational effects in 3-dimensional mechanical systems is an awkward process. This process can be considerably simplified by the more general "modeling by multiports" described in the next section. But only the latest developments of new modeling languages and simulator features make this multiport modeling approach practicable and powerful thus enabling the design engineer to exploit the potential of mathematical algorithms.

Physical systems may be classified into:

- Conservative physical systems: their quantities are flow quantities and difference quantities subject to compatibility constraints;
- Non-conservative physical systems: other interconnected systems consisting of elements with inputs and outputs and only one type of quantities.

A flow quantity (through quantity, 1-quantity) is measured at one point of the physical system (e.g. an electrical current or a hydraulic flow). A difference quantity (across quantity, 2-quantity) is measured between two points, e.g. a mechanical displacement or an electrical voltage. In Fig. 3 some of the most important flow and difference quantities are shown.

Physical domain	Flow quantity	Difference quantity
electrical	current	voltage
mechanical-translational	force	velocity
mechanical-rotational	torque	angular velocity
pneumatic	volume flow	pressure
thermal	heat flow	temperature

Fig. 3: Flow and difference quantities in different physical domains

The term "conservative physical systems" is motivated by the fact that time-independent compatibility constraints for flow and difference quantities are valid. In electrical systems, these constraints are the well-known KIRCHHOFF's current law (KCL, node law) and voltage law (KVL, mesh law). Similar conservation laws for flow and difference quantities hold also in

many other physical domains. This fact can be reflected in different classes of **system models**: conservative system models with conservative signals (flow and difference signals) and non-conservative systems (with non-conservative signals). We use the term "signal" as a model of all kinds of physical quantities which are able to exchange energy or information between the subsystems. We will use the term "network" to describe a system model consisting of elements, connections between these elements, and conservative signals. To stress the fact that these networks are models of physical systems in different domains, sometimes the term "generalized KIRCHHOFFian network" is used [41], [42].

Bond graphs [11], [43] are closely related to networks [30], but they are beyond the scope of this paper. From the author's point of view, the construction of bond graphs as models of real physical systems seems to be too complicated in many cases. The transformation of bond graphs into DAE systems is similar to the transformation of networks and can be done automatically.

Non-conservative quantities are typical of control systems or digital signal-processing devices. Block diagrams (see Fig. 2) or signal-flow graphs are appropriate models of these physical systems. In practice, control systems are often realized as electronic systems where voltages are the signals. Currents do not have any influence, and therefore, they are not included into the model. Conservation laws do not exist on this level of abstraction. From the practical point of view, the distinction between "analog" (time-continuous) and "digital" (time-discrete, mostly also value-discrete) non-conservative signals is very important. For the simulation of discrete (or digital) systems powerful specialized algorithms are available, e.g. the discrete-event simulation algorithms [2].

2. The multiport approach

The multiport modeling approach is illustrated in Fig 4.

- The complete system is decomposed into subsystems, the so-called multiports. Signals are separated into *external* signals between the subsystems, and *internal* signals within the subsystems. The external signals of a subsystem form its *terminal signals*.
- External signals may be conservative (flow signals e5, e6, e7, e8, e9, difference signals e1, e2 in the example) or non-conservative (e3, e4).
- The behavior of subsystems depends only on their terminal signals (and some internal signals) and their initial conditions, too.

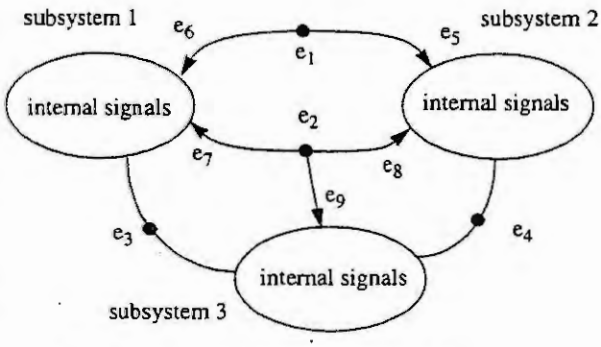


Fig. 4: Multiports in a system model

In electrical network theory, *multipoles* and *multiports* may be distinguished as such subsystem models. But in our more general context this distinction is not substantial, and therefore, we will use only the term multiport. Each subsystem can be substituted by other subsystems with the same terminal behavior [29] without any influence on the behavior of the rest of the system. The behavior of a subsystem can be expressed

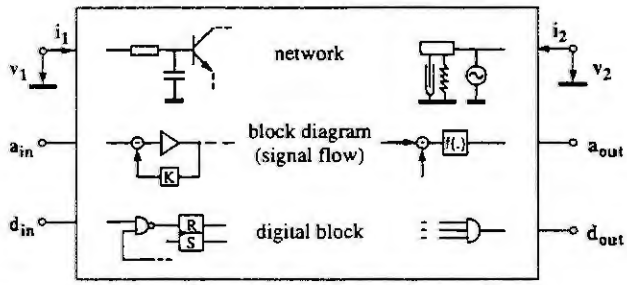
- implicitly by the connection of some other subsystems (or primitives on the lowest level): hierarchical structural refinement,
- explicitly by a set of equations (e.g. nonlinear differential-algebraic equations): "behavioral modeling" in a strict sense,
- or a combination of both if the simulator has language constructs to formulate mixed structural-behavioral descriptions.

The key problem in setting-up the system equations is the description of the terminal behavior of the *subsystems*. All other equations are essentially constraints resulting from the *connection* of the subsystems, especially the node and mesh law for flow and difference signals on conservative terminals, and can be constructed automatically.

An external view of a multiport is shown in Fig. 5. Terminal signals of the same type are assembled into vectors. So these signals are vector-valued functions of time t. The terminal signals are divided into the following categories:

v_1, i_2, a_{in}, d_{in}	independent by chooseable difference, flow, and non-conservative signals (analog and digital)
$v_2, i_1, a_{out}, d_{out}$	dependent difference, flow, and non-conservative signals (analog and digital)

In many cases it is impossible to state the terminal behavior only based on terminal signals. Subsystems may have internal states, and therefore, the introduction of additional signals s is necessary. The terminal behavior is determined also by the values of some parameters described by a parameter vector p. On very wide assumptions, the terminal behavior of multiports (Fig. 5) can be given by the following equations:



$$\mathbf{i}_1 = f_1(v_1, \dot{v}_1, i_2, \dot{i}_2, a_{in}, \dot{a}_{in}, s, \dot{s}, d_{in}, p, t) \quad (1)$$

$$v_2 = f_2(v_1, \dot{v}_1, i_2, \dot{i}_2, a_{in}, \dot{a}_{in}, s, \dot{s}, d_{in}, p, t) \quad (2)$$

$$a_{out} = f_3(v_1, \dot{v}_1, i_2, \dot{i}_2, a_{in}, \dot{a}_{in}, s, \dot{s}, d_{in}, p, t) \quad (3)$$

$$0 = f_4(v_1, \dot{v}_1, i_2, \dot{i}_2, a_{in}, \dot{a}_{in}, s, \dot{s}, d_{in}, p, t) \quad (4)$$

$$d_{out} = f_5(v_1, \dot{v}_1, i_2, \dot{i}_2, a_{in}, \dot{a}_{in}, s, \dot{s}, d_{in}, p, t) \quad (5)$$

Fig. 5: Terminal signals of a multiport

Eqs. (1) ... (4) are differential-algebraic equations (DAE). They can be solved by the numerical solution algorithms of a continuous system simulator [8]. The last equation (5) is mostly handled by discrete-event simulation algorithms which are very popular in general system simulation and in digital electronic simulation [1], [2], [25]. The *combined* treatment of continuous and discrete equations is known as hybrid simulation, analog-digital simulation, or mixed-signal simulation.

This multiport modeling approach is closely related to **object-oriented modeling**. Unfortunately, there is no generally accepted understanding of object-orientation in modeling and simulation. From the author's point of view, the following classification is useful:

- Object-oriented modeling: the construction of strictly **hierarchical, modular models.** This interpretation of object-oriented modeling was stressed by Cellier et al. [6], [7], [27].
- Object-oriented simulation: each subsystem is considered as an object. Each object has its own implemented simulation algorithm (a „method"). All objects communicate via message passing, coordinate their behavior, and so the simulation of the entire system is carried out [5]. A similar definition is: object-oriented simulation is the concurrent operation of different simulators without a global controller or a master simulator.
- Object-oriented programming: the application of programming languages like C++ or Smalltalk.

All these approaches exist independently from each other but they can be combined. Therefore, some authors [3], [17], [33] emphasize that object-oriented programming is not only the application of object-oriented programming languages but also a general method to design complex software systems. Hence, the method is very useful in developing complex models too.

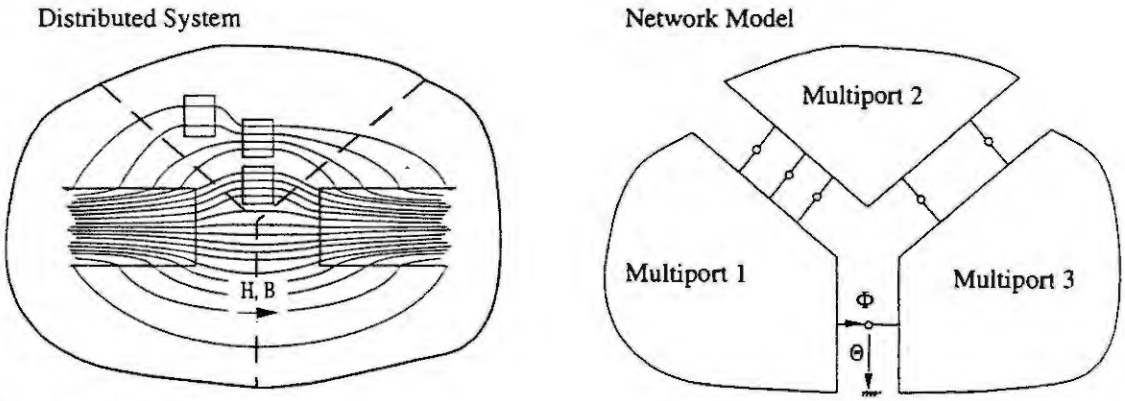
We will focus on *object-oriented modeling*. *Modularity* is guaranteed by the multiport approach because the interaction between the subsystems occurs only by the signals on the terminals (the interface). There are no global variables, and side-effects are excluded. *Hierarchy* is achieved by the structural refinement mentioned above and by the hierarchy concepts of programming languages or HDL used for behavioral modeling. Other aspects of modular modeling are

- The structural similarity between a functional-oriented partitioning of the original physical system and the decomposition of the system model into subsystem models.
- The user has only to model the behavior of subsystems (in a hierarchical description: on a lower level) and has to describe topologically the interconnection of the subsystems. Setting-up the equations describing the whole system is the task of the simulator (or its input language processor), and has not to be done by the user.
- With the same terminal description (interface) different behavioral descriptions may be used to obtain appropriate degrees of accuracy and multi-level-simulation.

Object-oriented programming languages [17], [33] are characterized by the concepts of data encapsulation and information hiding, message passing, a class concept and instantiation of classes, inheritance for building new classes, states and „methods" for changing the states. The application of these concepts allows to support substantially the correct construction of modular behavioral descriptions in different physical domains [12], [13], [14]. Some modern Hardware Description Languages [34], [24] and other modeling languages [23] are object-oriented.

3. Reduction of distributed systems to lumped systems

System partitioning and the multiport approach are described in Fig. 4 and Fig. 5 for systems with concentrated (lumped) elements only. However, many physical systems have spatially extended components, and fields have to be considered. Then, additional modeling steps are required to apply the multiport concept. In Fig. 6 these steps are illustrated. The physical system is partitioned in such a way that field regions are transformed onto connections. By integration over these field regions, the field quantities are transferred to signals carried by connections and terminals, respectively. Conservation laws hold in electrical, magnetic, thermal, ... fields and, therefore, conservative signals have to be used.



- Partitioning
- Definition of terminals
- Definition of concentrated flow and difference quantities (instead of field quantities)

Fig. 6: Distributed and concentrated systems

In practice, discretization methods are used to construct network models [28]. By spatial discretization, a PDE may be transformed into a set of ODEs which can be interpreted as a connection of network elements or multiports. In Fig. 7, a well-known example is presented.

$$\frac{\partial T}{\partial t} = \sigma \cdot \frac{\partial^2 T}{\partial x^2} \approx \sigma \cdot \frac{T(t, x_{k+1}) - 2T(t, x_k) + T(t, x_{k-1})}{(\Delta x)^2}$$

$$\frac{(\Delta x)^2}{\sigma} \cdot \frac{\partial T_k}{\partial t} = C \cdot \dot{T}_k = T_{k+1} - 2T_k + T_{k-1}$$

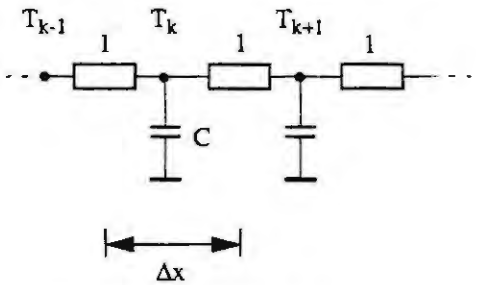


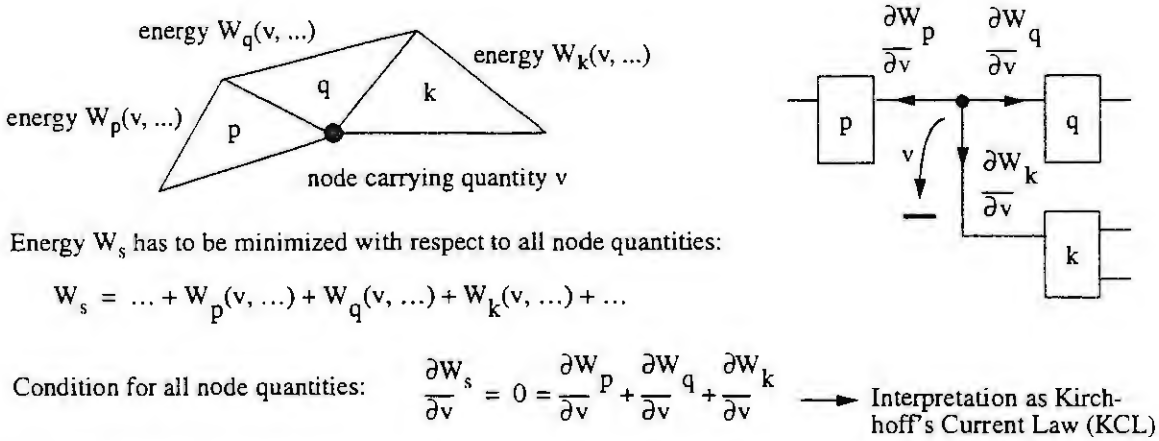
Fig. 7: Network interpretation of a discretized partial difference equation (thermal conduction)

Another modeling approach is the combination of the multiport concept with the **Finite Element Method** (FEM). Both methods are similar: partitioning the system and introducing "nodes" as carriers of flow and difference quantities. The basic idea of the Finite Element Method [4], [22], [44]:

- definition of an energy functional of the whole system as the sum of the functionals of the subsystems,
- minimizing this energy functional by appropriate choice of the node signals,
- choice of approximating „shape functions" for the quantities inside of the subsystems

can be applied to the construction of behavioral models of subsystems which can be connected to form a network [35], [26], [32], [36], [37]. In Fig. 8 the approach is illustrated generally.

In Fig. 9 a mechanical beam is choosen as an example. The static behavior of the beam with respect to global coordinates (x, y, z) is given by the equation in Fig. 9. C is the (3*3) transformation matrix between the local coordinate system (l, m, n) and the global system (x, y, z), E is the modulus of elasticity, I_m and I_n are the planar moments of inertia [26], [22].



Energy W_s has to be minimized with respect to all node quantities:

$$W_s = \ldots + W_p(v, \ldots) + W_q(v, \ldots) + W_k(v, \ldots) + \ldots$$

Condition for all node quantities: $\frac{\partial W_s}{\partial v} = 0 = \frac{\partial W_p}{\partial v} + \frac{\partial W_q}{\partial v} + \frac{\partial W_k}{\partial v}$ ⟶ Interpretation as Kirchhoff's Current Law (KCL)

Fig. 8: FEM approach and multiports

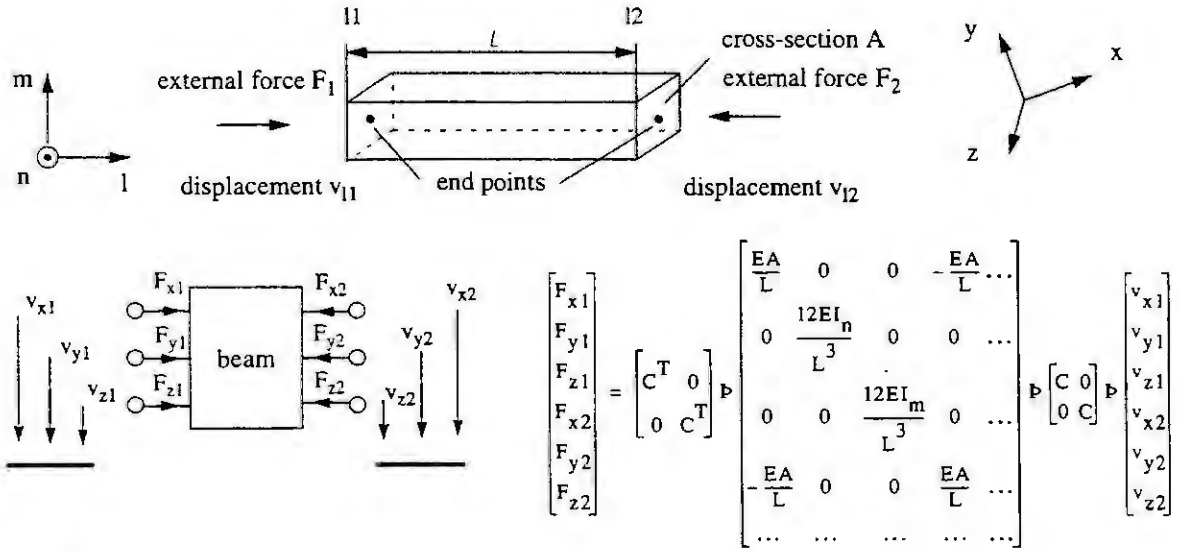


Fig. 9: Behavioral model of a beam element

The multiport modeling approach could be applied very successfully in modeling complex micromechanical sensors [26], [19], [38]. Fig. 10 shows the microphotograph of an acceleration sensor, Fig. 11 illustrates the modeling approach (partitioning into subsystems, modeling the subsystem behavior by FEM methods as multiports, interconnecting all multiports to model the entire system). In Fig. 12, a rotational sensor and the corresponding multiport network with conservative as well as non-conservative signals is presented.

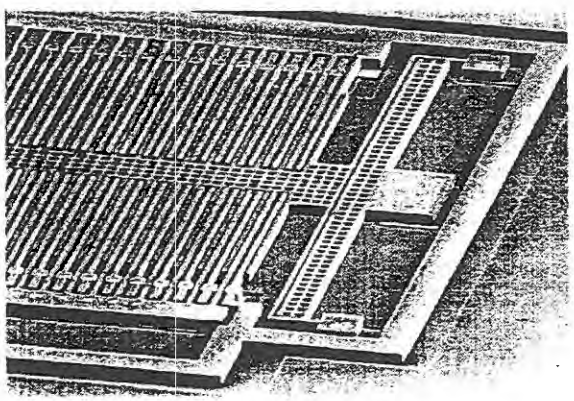
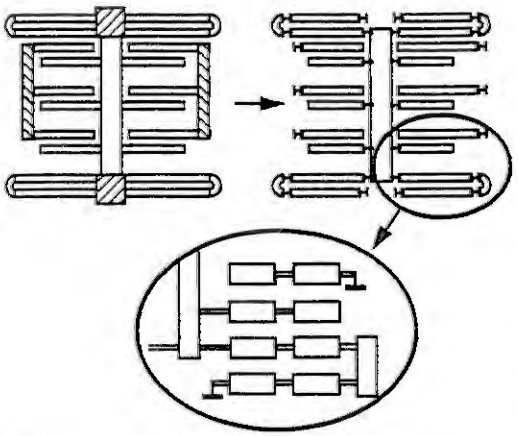
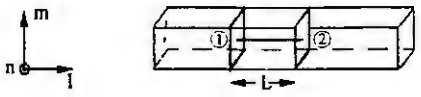


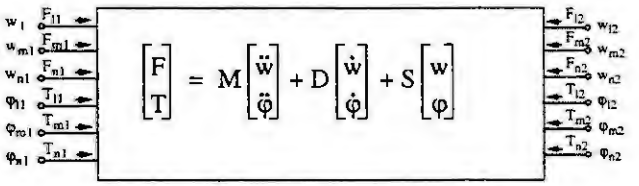
Fig. 10: Translational accelaration sensor



Decomposition into network elements



Flow and difference quantities:
force F, torque T
displacement w, rotation φ



FEM-based multi-terminal beam model

Fig. 11: Multiport model of the translational acceleration sensor

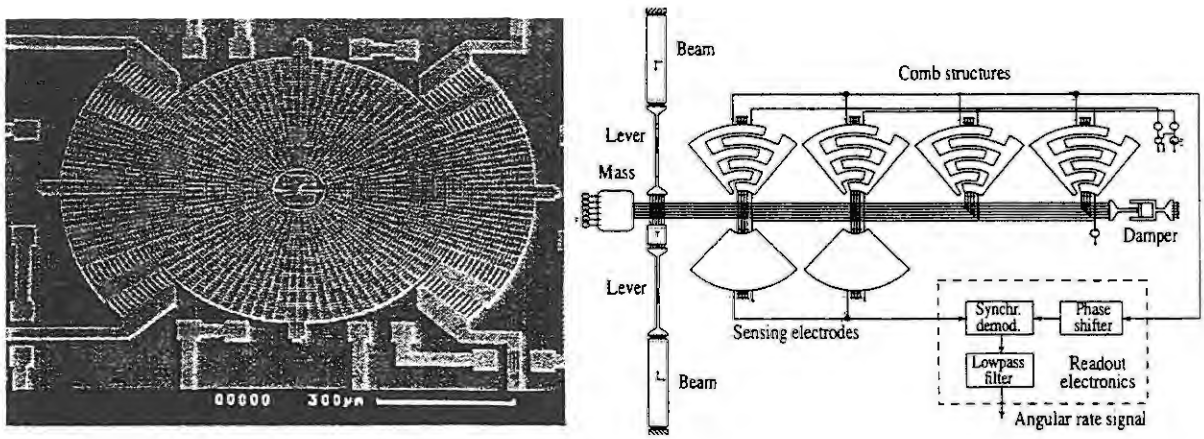


Fig. 12: Rotational acceleration sensor and its multiport model

4. Modeling Languages

If the design engineer is able to formulate directly the model equations then they can be solved with programs like Mathematica, Maple, Matlab, MatrixX or ACSL. Graphical front ends (Simulink, SystemBuild) are important tools to capture the structure of the system. Another approach is the application of simulator-independent modeling languages (in electronics: Hardware Description Languages, HDL). Modern languages like VHDL-AMS [21], [40] and Modelica [23] support all these aspects of systems and signals discussed above:

- multi-domain,
- conservative and non-conservative,
- continuous and discrete (mixed-signal),
- hierarchical structural and behavioral description.

They have been provided with features for the precise definition of terminals at the subsystem interfaces and of the physical nature of quantities and signals. Hierarchical and modular modeling is assisted, in addition Modelica is an object-oriented language. VHDL-AMS is under IEEE standardization and will be supported by several system simulators in the near future. At the moment, Modelica is accepted only by a prototype simulator developed on the Dymola simulator [9] basis.

```
ENTITY entity_name IS
        GENERIC     ( p : ... );                                          -- description of the
        PORT        (TERMINAL t1: ... ; QUANTITY aout :... ; SIGNAL din: ... );-- subsystem interface
END ENTITY entity_name;

ARCHITECTURE name OF entity_name IS
        QUANTITY v1 ACROSS i1 THROUGH t1;        -- declaration of terminal quantities
        QUANTITY s : ... ;                       -- additional states,
        CONSTANT ...;                            -- constants, ...

BEGIN
        relations in accordance to equations (1), (2),( 3) for
                the dependent terminal flow quantities i1,
                the dependent terminal difference quantities v2, and
                the output quantities aout

        formulation of equ. (4) to force f4 to zero

        calculation of dout in a VHDL-like manner

END ARCHITECTURE name;
```

Fig. 13: Skeleton of a VHDL-AMS description of the multiport behavioral equations (in relation to Fig. 5)

In Fig. 13 the structure of a VHDL-AMS description of the multiport behavioral equations (1) ... (5), see also Fig. 5, is shown. The interface description (ENTITY) is strictly separated from the behavioral description (ARCHITECTURE). It is allowed to combine different ARCHITECTURE bodies with one ENTITY. In this way it is possible to use models with different levels of abstraction and of numerical accuracy by exchanging the ARCHITECTURE part without any influence on the structure of the whole system.

The impressive examples of Figs. 11 and 12 illustrate the power of the generalized network modeling approach but its practicability depends strongly on the modeling language available in modern system simulators.

Acknowledgement. I would like to thank my colleagues in the modeling and simulation group at FhG EAS Dresden and Kurt Reinschke for many years of fruitful cooperation as well as Karl-Heinz Diener, Joachim Haase, Roland Jancke, and Gunter Kurth for their assistance in preparing this paper. The funding by the German Ministry of Research (BMBF) in different projects and in DFG-SFB 358 „Automated System Design" is also gratefully acknowledged.

References

1. Armstrong, J.R.: Chip-Level Modeling with VHDL. Prentice Hall, Englewood Cliffs 1989.
2. Banks, J. (Ed.): Handbook of Simulation. Wiley, New York 1998.
3. Berge, J.-M.; Levia, Oz; Rouillard, J.: Object-Oriented Modeling. Kluwer, Dordrecht 1993.
4. Braess, D.: Finite Elemente. Springer, Berlin 1997.
5. Buck, J.T.; Ha, S.; Lee, E.A.; Messerschmidt, D.G.: Ptolemy: a framework for simulating and prototyping heterogeneous systems. Int. J. Computer Simulation 4(1994) April, 155-182.
6. Cellier, F. E.: Continuous System Modeling. Springer, New York/Berlin 1991.
7. Cellier, F. E.; Elmquist, H.; Otter, M.: Modeling from physical principles. In: Levine, W.S. (Ed.): The Control Handbook. CRC Press, Boca Raton, FL, 1996, 99-107
8. Clauß, C.; Haase, J.; Kurth, G.; Schwarz, P.: Extended admittance description of nonlinear n-poles. Archiv Elektronik und Übertragungstechnik 40(1995)2, 91-97.

9. Elmquist, H.: Dymola - Dynamic Modeling Language, Users manual. Dynasim AB, Lund 1994. See also: http://www.dynasim.se

10. Firestone, F.A.: The mobility method for computing the vibration of linear mechanical and acoustical systems: mechanical-electrical analogies. J. Applied Physics 9(1938) June, 373- 387.

11. Karnopp, D. C.; Margolis, D. L.; Rosenberg, R. C.: System Dynamics: A Unified Approach. Wiley, New York 1990.

12. Kasper, R.; Koch,W.: Object-oriented behavioral modelling of mechatronic systems. Proc. 3rd Conf. Mechatronics and Robotics, Paderborn 1995, 70-84

13. Kecskeméthy, A.: Objektorientierte Modellierung der Dynamik von Mehrkörpersystemen mit Hilfe von Übertragungselementen. VDI-Fortschrittsberichte Reihe 20, Nr. 88, Düsseldorf 1993.

14. Kecskemethy, A.; Hiller, M.: An object-oriented approach for an effective formulation of multibody dynamics. 2nd US Natl. Congress Computational Mechanics, Washington, 1993.

15. Klein, A.; Gerlach, G.: System modelling of microsystems containing mechanical bending plates using an advanced network description method. Proc. MICRO SYSTEM Technologies, Potsdam 1996, 299-304.

16. Koenig, H. E.; Blackwell, W. A.: Electromechanical System Theory. McGraw-Hill, New York 1961.

17. Kowalk, W.P.: System - Modell - Programm. Spektrum Akademischer Verlag, Heidelberg 1996.

18. Lenk, A.: Elektromechanische Systeme (3 vol.). Verlag Technik, Berlin 1971 - 1973.

19. Lorenz, G.; Neul, R.: Network-type modeling of micromachined sensor systems. Proc. MSM98.

20. MacNeal, R.H.: The solution of elastic plate problems by electrical analogies. J. Applied Mechanics 18(1951)1, 59-67.

21. Mantooth, H. A.; Fiegenbaum, M. F.: Modeling with an Analog Hardware Description Language. Kluwer, Dordrecht 1994.

22. Meissner, U.; Menzel,A.: Die Methode der Finiten Elemente. Springer, Berlin 1989.

23. Modelica: see http://modelica.org ; many links to Modelica-related publications.

24. Müller, W.; Rammig, F.: ODICE: Object-oriented hardware description in CAD environment. Proc. 9. Conf. Computer Hardware Description Languages (CHDL'90), Elsevier, Amsterdam 1990, 19-34.

25. Navabi, Z.: VHDL – Analysis and Modeling of Digital Systems. McGraw-Hill, New York 1993.

26. Neul, R. et al.: A modeling approach to include mechanical microsystem components into system simulation. Proc. Design, Automation & Test Conf. (DATE'98), Paris, 1998, 510-517.

27. Otter, M.: Objektorientierte Modellierung mechatronischer Systeme am Beispiel geregelter Roboter. Dissertation, Bochum 1994.

28. Pelz, G. et al.: MEXEL: Simulation of microsystems in a circuit simulator using automatic electromechanical modeling. Proc. MICRO SYSTEM Technologies, VDE-Verlag, Berlin 1994, 651-657.

29. Reibiger, A.: On the terminal behaviour of networks. Proc. ECCTD '85, Prague, September 1985, 224-227.

30. Reibiger, A.; Loose, H.: Bond graphs and matroids. (these Proceedings).

31. Reinschke, K.; Schwarz, P.: Verfahren zur rechnergestützten Analyse linearer Netzwerke. Akademie-Verlag, Berlin 1976.

32. Romanowicz, B. F.: Methodology for the Modeling and Simulation of Microsystems. Kluwer, Dordrecht 1998.

33. Rumbaugh, J. et al.: Object-Oriented Modeling and Design. Prentice Hall, Englewood Cliffs, 1991.

34. Schumacher, G.; Nebel, W.: Survey of languages for object-oriented hardware design methodologies. In: Berge, J.-M. et al. (Eds.): High-Level System Modeling: Specification Languages. Kluwer, Dordrecht 1995.

35. Schwarz, P.; Haase, J.: Behavioral modeling of complex heterogeneous microsystems. Proc. 1st Intern. Forum on Design Languages (FDL'98), Lausanne, Sept. 1998, 53-62.

36. Senturia, S. D.: CAD Challenges for Microsensors, Microactuators, and Microsystems. Proc. IEEE 86(1998)8, 1611-1626.

37. Senturia, S.; Aluru, N. R.; White, J.: Simulating the behavior of MEMS devices: computational methods and needs. IEEE Trans. Computational Science & Engineering, January 1997, 30-54.

38. Teegarden, D.; Lorenz, G.; Neul, R.: How to model and simulate microgyroscopic systems. IEEE Spectrum 35(1998)7, 67-75.

39. Tonti, E.: The reason for analogies between physical theories. Appl. Math. Modelling 1(1976), 37-50.

40. VHDL-AMS (VHDL – Analog and Mixed Signal Extensions): see http://www.vhdl.org/analog/ .

41. Voigt, P.; Wachutka, G.: Electro-fluidic microsystem modeling based on Kirchhoffian network theory. Sensor and Actuators A 66 (1998)1-3, 6-14.

42. Wachutka, G.: Tailored modeling: a way to the 'virtual microtransducer fab' ? Sensor and Actuators A 46-47 (1995), 603-612.

43. Wellstead, P.E.: Introduction to Physical System Modelling. Academic Press, London 1979.

44. Zienkiewicz, O. C.; Taylor, R. L.: The Finite Element Method (2 vol.). McGraw-Hill, New York 1989 and 1991.

MODELLING AND SIMULATION OF COMBINED LUMPED AND DISTRIBUTED SYSTEMS BY AN OBJECT-ORIENTED APPROACH

C. Maffezzoni and M.L. Aime
Politecnico di Milano, Dipartimento di Elettronica e Informazione
Piazza Leonardo da Vinci 32, 20133, Milano, Italy

Abstract The paper extends object-oriented modelling (OOM) to physical systems including distributed parameter components. An approach is proposed where the basic paradigms of OOM (e.g. encapsulation, independence of module interface from internal behaviour, non-causal model format) are ensured. The technical aspects are worked out with reference to the case of heat exchangers (HEs) of generic topology. A possible implementation in the OOM language Modelica is outlined, to prove the compliance of the extension with OOM constructs.

Introduction

OOM is a widely accepted technique which has already produced both modelling languages [1], [2], [3] and actual software packages [4], [5]. The approach is based on a number of paradigms, among which the following ones certainly play a fundamental role:

- The definition of physical ports (even called terminals) as the standard interface to connect a certain component model to the rest of the world, to reproduce the component structure of the physical system
- The definition of models in a non-causal form so as to allow reuse, abstraction and unconditional connection
- The independence of the model interface (physical ports) of the internal description

The state of the art of OOM is well represented by the development of the Modelica project [2], a recent international effort to define a standard language: there are well assessed methods to treat Lumped Parameter Components (LPC); there are not, on the contrary, unified solutions to describe Distributed Parameter Components (DPC), while respecting the fundamental paradigms listed above. Indeed, the normal way (see e.g. [6]) to build models of DPC is to split the physical domain into finite subvolumes and to introduce fictitious physical ports which connect those subvolumes, describing the behaviour of any piece of the system by differential algebraic equations (DAEs). In this way components' interfaces are still point-wise physical ports (like in LPC) and the model description presents several independent ports which do not exist in the reality: those ports are actually part of the numerical method (finite-volumes, finite-difference, etc.) employed to solve the partial differential equation (PDE) model. This destroys the one-to-one correspondence between the structure of the physical system and that of the model, violates basic principles of OOM, leading to a substantial loss of modularity.

Aim of the present paper is to extend the OOM approach to deal with physical systems including DPCs, adopting for them the very same paradigms as for the case of LPC systems, keeping the basic principle that components interface must be defined independently of the internal model description and that possible discretization methods used to approximate PDE must not be relevant in the model structuring. The proposed solution is based on the introduction of physical interface that extend over a finite spatial domain (the so-called distributed terminals) and the assumption of a class of numerical methods (like finite element method (FEM) with weak formulation of boundary conditions) which keep the model in non-causal form. It is worth mentioning that the simulation package gPROMS [5] is equipped to deal with PDEs while respecting the distributed nature of the boundary conditions and the separation between interface structure and numerical solution method. The main difference comes from the fact that gPROMS adopts a model description which is based on equations and boundary conditions instead of on components and terminals, even though it well supports modularity. The presentation is focused on the case of HEs (sections 2 and 3), while generalisations are considered in the conclusions.

Example: Heat Exchangers

A general scheme of a HE is shown in Fig. 1; its mathematical model is built by mass, momentum and energy balance equations. For simplicity, only the energy conservation equations for the overall HE are presented and discussed here, without loosing the generality of the discussion. The energy balance equations for the internal and

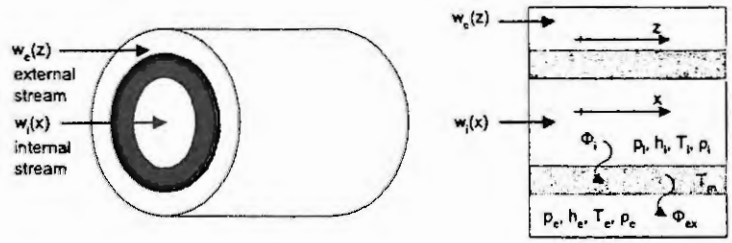


Figure 1: A general scheme of a heat exchanger

external flow stream, and for the metal wall are [7]

$$\rho_{fi}A_{fi}\frac{\partial h_{fi}}{\partial t} + w_{fi}\frac{\partial h_{fi}}{\partial z} - A_{fi}\frac{\partial p_{fi}}{\partial t} - A_{fi}u_{fi}\frac{\partial p_{fi}}{\partial z} - \frac{C_{f\_fi}}{2}\rho_{fi}\varpi_{fi}u_{fi}^2|u_{fi}| = \Phi_i \tag{1}$$

$$\rho_{fe}A_{fe}\frac{\partial h_{fe}}{\partial t} + w_{fe}\frac{\partial h_{fe}}{\partial x} - A_{fe}\frac{\partial p_{fe}}{\partial t} - A_{fe}u_{fe}\frac{\partial p_{fe}}{\partial x} - \frac{C_{f\_fe}}{2}\rho_{fe}\varpi_{fe}u_{fe}^2|u_{fe}| = \Phi_{ex} \tag{2}$$

$$\rho_m c_m \sigma_m \frac{\partial T_m}{\partial t} = -\Phi_i + \Phi_{ez} \tag{3}$$

where

$$\Phi_i(z,t) = \gamma_i\left[T_{fi}(z,t) - T_m(z,t)\right] \tag{4}$$

$$\Phi_{ex}(x,t) = \int_0^H \Psi_e(x,z,t)dz \tag{5}$$

$$\Phi_{ez}(z,t) = \int_0^L \Psi_e(x,z,t)dx \tag{6}$$

$$\Psi_e(x,z,t) = \gamma_e S_e\left[T_m(z,t) - T_{fe}(x,t)\right]\Gamma_{conv}(x,z) \tag{7}$$

for $0 < x \leqslant L$, $0 < z \leqslant H$, $t > 0$. The initial (Dirichlet) conditions are $h_{fi}(z,0) = h_{fi_0}(z)$ and $T_m(z,0) = T_{m_0}(z)$ for $0 \leqslant z \leqslant H$, and $h_{fe}(x,0) = h_{fe_0}(x)$ for $0 \leqslant x \leqslant L$. The boundary conditions are $h_{fi}(0,t) = h_{fi_{in}}(t)$ and $h_{fe}(0,t) = h_{fe_{in}}(t)$, for $t > 0$, where $h_{fi_0}(z)$ and $h_{fe_0}(x)$ are the initial internal and external enthalpy and $T_{m_0}(z)$ is the initial wall temperature. The meaning of the variables is: h is the enthalpy, T_m the wall temperature, ρ_m the wall density, c_m the wall heat capacity, σ_m the wall thickness. S_e is the external wall surface area, γ_i and γ_e are the internal and external convective heat transfer coefficient. Φ_i is the internal convective heat transfer, $\Psi_e(x,z,t)$ is the heat flow distribution over the entire HE, and Φ_{ez} and Φ_{ex} are the heat flows along z and x, respectively. $\Gamma_{conv}(x,z)$ is the form factor of the convective heat exchange. z and x are the axes along the tube (or internal) and shell (or external) flow stream. Clearly all the internal fluid properties (flow speed, density, temperature, enthalpy, etc.) are functions of x and t. Instead the external stream ones vary with t and in the z direction as the flow. L and H are the overall lengths of the tube and shell. The subscripts 'fi' and 'fe' stand for the internal and external stream properties.

It is necessary to assume that $\int_0^L \int_0^H \Gamma_{conv}(x,z)dzdx = 1$, so that the overall heat $Q(t)$ exchanged between the internal and external stream is

$$Q(t) = \int_0^L \int_0^H \Psi_e(x,z,t)dzdx \tag{8}$$

The presented model and approximation are valid for any type of HE topology (cross-flow or counter-cross flow or long-flow). The difference between the various kinds of HE is expressed by the different formulas of $\Gamma_{conv}(x,z)$. For this presentation, only the case of a cross-flow convective HE is considered (see Fig. 2), where the factor form is expressed by

$$\Gamma_{conv\_cr}(x,z) = \frac{1}{LH}\delta\left(\frac{x}{L} - f\left(\frac{z}{H}\right)\right) \tag{9}$$

where δ is the Dirac delta function and $f(\cdot)$ is the function describing the tube layout. If $\Gamma_{conv\_cr}(x,z)$ is expressed

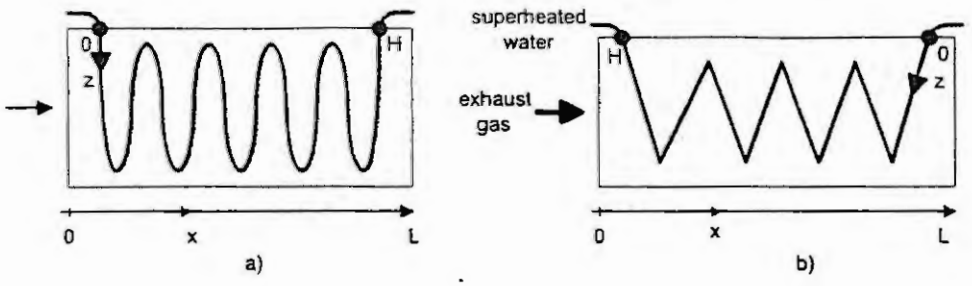


Figure 2: Cross-flow heat exchanger: a) physical topology; b) scheme of the assumed geometry

by (9), $\Phi_{ex}(z,t)$ and $\Phi_{ex}(x,t)$ have the following expressions

$$\Phi_{ex}(z,t) = \gamma_e \varpi_e \left[T_m(z,t) - T_{fe}(Lf(\frac{z}{H}),t)\right] \tag{10}$$

$$\Phi_{ex}(x,t) = \frac{H}{L}\gamma_e \varpi_e \left.\frac{df}{d\alpha}\right|_{\alpha=f'(\frac{x}{L})} \left[T_m(Hf'(\frac{x}{L}),t) - T_{fe}(x,t)\right] \tag{11}$$

where $f'(\cdot)$ is the inverse of $f(\cdot)$. For instance, for the case of Fig. 2a $f(\cdot) = \sin(\cdot)$, while for Fig. 2b, where the function $f(\cdot)$ is supposed to be linear between two tube turns, $f(\frac{z}{H}) = \frac{z}{H}$, $f'(\frac{x}{L}) = \frac{x}{L}$, $\frac{df}{d\alpha} = 1$.

Object-Oriented Model Structure

The object-oriented (OO) model of the HE shown in Fig. 2 is depicted in Fig. 3. In the OOM external representation (Fig. 3a), the HE is connected through lumped physical terminals to the external environment. In particular a couple of the terminals (TT_1-TT_2) is used to connect the internal stream to the other components, and the other couple (TT_3-TT_4) is used by the external stream. However the internal representation uses the dHT terminals in order to connect the internal and external stream to the wall (Fig. 3b). In addition, the wall OOM is a composite model, internally represented by other three sub-models: the internal and external boundary layer, and the metal wall (Fig. 3c). Each module is associated with the equations (1)-(7), (9) representing the HE mathematical model.

The lumped physical terminals are associated with the following variables: $TT_1 = (h_1, w_1, p_1)$, $TT_2 = (h_2, w_2, p_2)$, $TT_3 = (h_3, w_3, p_3)$, $TT_4 = (h_4, w_4, p_4)$, where w_i, p_i, for $i = 1, \ldots, 4$ are the stream flow rate and pressure. Any distributed terminal represents a distributed interface and consists of a couple of effort (temperature) and flow (heat flux) profiles: $dHT_1 = (\mathbf{T}_{fi}, \Phi_i)$, $dHT_2 = (\mathbf{T}_m, \Phi_i)$, $dHT_3 = (\mathbf{T}_m, \Phi_{ex})$, $dHT_4 = (\mathbf{T}_{fe}, \Phi_{ex})$, where $\mathbf{T}_{fi}$, $\mathbf{T}_{fe}$ and $\mathbf{T}_m$ are the internal and external stream, and wall temperatures.

Module Formulation in Non-Causal Form

The HE mathematical model is a system of PDEs (1)-(7), (9), whose approximated solution is obtained through the application of appropriate numerical methods. In the present work we focus on the FEM since it provides advantages respect to other approximation methods. Various FEMs can be found in literature, but the stabilization methods for the Galerkin FEM [8], [9], [10] are recommended for advection problems as (1), (2). The boundary conditions can be imposed in two different ways: strong and weak formulation [9]. The weak formulation is chosen here.

Thanks to the chosen approximation method, the causality is not fixed in the HE module, i.e. the model structure is independent of the flow direction, since only a different set of DAEs is used in the case of flow reversal. Moreover, the resulting DAE system has always the same state-variables (nodal values) since the nodal values on the boundary do not coincide with the boundary conditions. Since the model structure is independent on the flow direction, the terminals (both the lumped and distributed ones) are independent too. Therefore the module is really non-causal.

The new concept of distributed terminals is introduced in order to represent a distributed system using a more general and complete approach. Basically, a distributed terminal consists of a set of vector variables. Clearly it is possible to link only distributed terminals having the same length of the vector, i.e. the same discretization of the distributed variables. However, in order to be as flexible as possible, the internal space discretization can in general

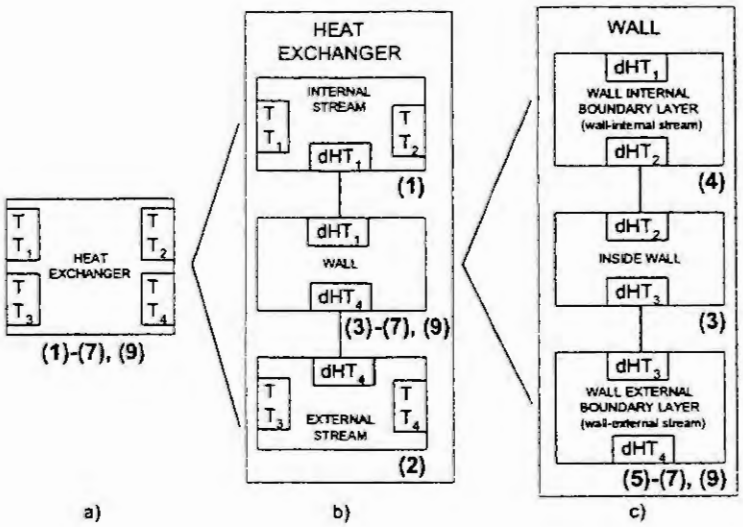


Figure 3: Object-oriented modelling of a heat exchanger: a) HE external OOM representation; b) HE internal OOM representation; c) wall internal OOM representation

be different from the terminal variable discretization. Clearly some kind of interpolation would be then necessary to transform one set of variables into the other.

The particular representation chosen for the HE has several advantages. First, the HE mathematical model is valid for any type of HE topology, which is expressed by the different formulas of $\Gamma_{conv}(x, z)$. Second, the interaction between physical components, i.e. internal and external stream and inside wall, is represented and confined in the "interaction" modules, i.e. the wall internal and external boundary layers. In this way, the OOM of the HE allows to standardize the dHT terminals.

Comparison to Subunit Model Representation

The OOM approach of DPC through subunits means that the HE can be represented as shown in [6]. The overall HE is divided into sections (control volumes) described by a DAE system. Therefore the interaction between the internal and external stream and the wall is described through point-wise terminals relative to the different control volumes. It is important to note that, in this case, the HE topology, i.e. the relationship between the two spatial domains x and z, is taken into account while writing the approximation through the finite volume method. In other word, the coupling between x and z is expressed by the coupling of the subunit models. The main drawback is that the heat exchange interface structure is built as part of the numerical method chosen for the solution approximation.

Possible Implementation in an Object-Oriented Language

The proposed methodology is formalized in the new standard OOM language Modelica, adding specific syntax elements for the new structures, such as the distributed heat transfer terminal, which more closely represent the physical interaction occurring over a spatially distributed domain. Omitting all the details concerning the internal model description by FEM for the sake of brevity, only the new structure dHT is introduced. Using the standard Modelica syntax, the distributed terminal dHT can be defined as:

Connector dHTTerminal, Temperature $\mathbf{T}$; HeatFlux Φ; end dHTTerminal.

T and Φ are vector variables of the same length. When connecting two distributed terminals, the set of vector variables are forced to be equal (effort variable) or to be opposite (flow variable). Thanks to the new syntax the HE OOM can be implemented in the Modelica language. Of course, the model has to be formulated in discretized form because Modelica does not directly support PDE approximation.

Conclusions

It has been shown how OOM can be successfully applied to the case of Distributed Parameters heat exchanging systems, possibly combined with generic Lumped Parameters Components. The main contribution of the paper is to define a model structuring approach in which distributed interfaces are modelled as distributed terminals which are fully standard in a given physical domain (e.g. heat-transfer terminals, consisting of a spatial effort function, the temperature profile, and a spatially flow function, the heat flux), so ensuring independence of the module interface from the internal model description. To do this, a basic point is to clearly distinguish the heat-transfer pattern (which is accounted for in the model of the boundary layer) from the connection links which are uniquely established between identical terminals (coincidence of effort variables, sum-to-zero of flow variables). In addition, systematic FEM approach with weak formulation of boundary conditions is proposed to ensure non-causal formulation of DPC models.

The concepts of distributed terminal and non-causal model are easily extendable to systems different from heat exchangers (consider, for instance, submarine cables or electromagnetic devices). The basic points are to clearly separate the interface/connection problem from the internal modelling format. In this respect, we must neatly distinguish the interaction modes which take place on a spatial domain (such as the heat transfer patterns through the thin layer existing between a fluid stream and a boundary wall), which are actually a special component of the distributed system in which peculiar physical phenomena occur, from the pure connection between two identical physical ports.

References

[1] S. Mattsson and M. Andersson, "OMOLA - an object-oriented modelling language," in *Recent Advances in Computer Aided Control Systems Engineering* (M. Jamshidi and C. Herget, eds.), Elsevier - New York, 1992.

[2] H. Elmqvist, S. Mattsson, and M. Otter, "Modelica - a language for physical system modeling, visualization and interaction," in *CACSD (IEEE Symposium on Computer-Aided Control System Design)*, (Hawaii), 1999.

[3] C. Maffezzoni and R. Girelli, "MOSES: Modular modelling of physical systems in an object-oriented database," *Mathematical Modelling of Systems*, vol. 4, no. 2, pp. 121–147, 1998.

[4] H. Elmqvist, F.E.Cellier, and M. Otter, "Object-oriented modelling of hybrid systems," in *ESS (European Simulation Symposium)*, (Delft - Holland), 1993.

[5] gPROMS, "Advanced user guide," tech. rep., Process Systems Enterprise, 1998.

[6] B. Nilson and J. Eborn, "Object-oriented modelling of thermal power plants," *Mathematical Modelling of Systems*, vol. 4, no. 3, pp. 207–218, 1998.

[7] M. Aime, *Engineering Methods and Tools for Modeling and Simulation of Power Generation Plants*. PhD thesis, Politecnico di Milano, 1999.

[8] D. Duncan and D. Griffiths, "The study of a Petrov-Galerkin method for first-order hyperbolic equations," *Computer Methods in Applied Mechanics and Engineering*, vol. 45, pp. 147–166, 1984.

[9] A. Quarteroni and A. Valli, *Numerical Approximation of Partial Differential Equations*. Springer-Verlag, 2nd ed., 1997.

[10] L. Franca, S. Frey, and T. Hughes, "Stabilized finite element methods: I. Application to the advective-diffusive model," *Computer Methods in Applied Mechanics and Engineering*, vol. 95, pp. 253–276, 1992.

Alternatives In The Generation Of Time Domain Models Of Fluid Lines Using Frequency Domain Techniques

Wayne J. Book
Georgia Institute of Technology
Atlanta, GA 30332-0405, U.S.A.

Cody Watson
Lockheed-Martin Co.
Fort Worth, TX U.S.A.

Abstract. By converting from frequency domain models to time domain models, nonlinear behavior and linear distributed behavior can both be effectively represented. Three methods are presented to convert fluid line models from the frequency domain to the time domain. Comparison shows that combination of components in the frequency domain has advantages in accuracy and efficiency in many practical cases. Methods of finding model poles and residues and ways to avoid numerical difficulties with poles at the origin are discussed.

Introduction.

Distributed components and systems are sometimes well described by linear partial differential equations and can thus be analyzed by frequency domain approaches. Frequency domain techniques allow the combination into a system model of linear components both distributed and lumped. One may prefer to convert the frequency domain model back to a finite order model for purposes of analysis, control or system design, or because additional components need to be included which are nonlinear and hence cannot be accurately modeled in the frequency domain.

Modeling pressure transients in fluid lines is an example of the general modeling task described above. This paper treats the modeling of fluid lines specifically but flexible structures [1] and other engineering systems are similar in their modeling needs. The fluid lines studied here might appear in a variety of engineering systems such as automobiles [3] and airplanes (fuel injection, brake lines, transmissions) and manufacturing equipment (hydraulic power lines).

This paper will present the techniques used to model a small number of components combined into a linear system. The overall process will be described but the focus of the paper will be on several issues that can form obstacles to the modeler. Comparisons will be made between three versions of this approach. Firstly, frequency domain component models can be immediately converted to time domain models, then combined with other time domain models. This is referred to as the Time Domain Combination (TDC) approach. The analytical conversion of the frequency domain model to the time domain produces the TDC-AC approach. Secondly, frequency domain models can be Numerically Converted to produce time domain approximations leading to the TDC-NC approach. Finally, the components can be combined in the frequency domain and numerically converted to the time domain (FDC-NC). Each approach has its own appeal. The modeling process was implemented in MATLAB © and Simulink © to obtain numerical and graphical support needed.

The paper will first describe briefly the components of the systems at hand. The fluid line is the only distributed component and will be treated in the most detail. Causality of the model becomes an important issue when components are combined. The conversion of the frequency domain component model will then be discussed. This technique has been explored in considerable detail in prior papers. Some new approaches have been applied to finding the residues, however. Then the frequency domain combination will be described. The numerical conversion (FDC-NC) will be discussed in detail. Finally, the results will be compared for a standard "water hammer" effect, the solution of which is analytically known for comparison. More detail is found in [8].

Fluid Line Component Models

Simple linear lumped parameter components, resistances, capacitances and inductances are important and result from ordinary differential or algebraic equations. The resistance, capacitance, and inductance, for example, are modeled as

$$\begin{bmatrix} P_a \\ Q_a \end{bmatrix} = \begin{bmatrix} 1 & R \\ 0 & 1 \end{bmatrix} \begin{bmatrix} P_b \\ Q_b \end{bmatrix} \quad \begin{bmatrix} P_a \\ Q_a \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ Cs & 1 \end{bmatrix} \begin{bmatrix} P_b \\ Q_b \end{bmatrix} \quad \begin{bmatrix} P_a \\ Q_a \end{bmatrix} = \begin{bmatrix} 1 & Ls \\ 0 & 1 \end{bmatrix} \begin{bmatrix} P_b \\ Q_b \end{bmatrix} \tag{1}$$

where:

P_a, P_b = upstream and downstream pressure, respectively
Q_a, Q_b = upstream and downstream flow, respectively
R = resistance value; C = capacitance value; L = inductance value

A distributed fluid line is considerably more complex, and based on partial differential equations. This paper uses the dissipative or "exact" model[3] to describe fluid flow in a line. For this model, the fluid equations governing the flow in a differential boundary within a cylindrical fluid line are:

Momentum $$\rho_0 \frac{\partial u}{\partial t} + \frac{\partial p}{\partial x} = \mu_0 \left[\frac{\partial^2 u}{\partial r^2} + \frac{1}{r} \frac{\partial u}{\partial r} \right] \quad (2)$$

Continuity $$\frac{\partial \rho}{\partial t} + \rho_0 \left[\frac{\partial u}{\partial x} + \frac{\partial v}{\partial r} + \frac{v}{r} \right] = 0 \quad (3)$$

State $$\frac{d\rho}{\rho_0} = \frac{dp}{K_B} \quad (4)$$

where:

ρ_0, μ_0 = average density, absolute viscosity
u, v = axial, radial velocity
x, r = axial, radial coordinates
θ = temperature
K_B = bulk modulus

And we assume line length/line radius >> 1; small density change: $\Delta\rho / \rho_0 << 1$; radius of curvature of the line > radius of line and non-turbulent mean flow. There are simpler fluid property models that may be used, such as the inviscid or frictionless model and the linear friction model.

Much research has been done on the approximation of the dynamic response of liquid filled lines using the dissipative model[3,4,5,6,9]. This paper uses the same starting point as a basis for approximating the dynamic characteristics of a line, namely, the frequency domain representation[7] shown in Equations 5-9. Equation 9 is a general frequency domain transfer function relationship of the input and output flows, where s is the Laplace variable, $\Gamma(s)$ is the propagation operator, and $Z(s)$ is the characteristic impedance. Equation 5 is general because selecting either the inviscid, linear friction, or dissipative model produces the same representation with different expressions for $\Gamma(s)$ and $Z(s)$. The $\cosh(\Gamma \ell)$ term is significant in that is represents the transmission of pressure or flow along the line. For instance $\cosh(\Gamma \ell)$ produces a pure time delay in the inviscid model, while in the dissipative model it produces a time delay with frictional losses. The characteristic impedance provides a relationship of the flow at a point in the line to the corresponding pressure at that point[3].

$$\begin{bmatrix} P_a(s) \\ Q_a(s) \end{bmatrix} = \begin{bmatrix} \cosh(\Gamma(s)) & Z(s)\sinh(\Gamma(s)) \\ \frac{1}{Z(s)}\sinh(\Gamma(s)) & \cosh(\Gamma(s)) \end{bmatrix} \begin{bmatrix} P_b(s) \\ Q_b(s) \end{bmatrix} \quad (5)$$

Equations 6-9 represent the operators that form the dissipative model[3,7].

$$D_n = \nu_0 L / c_0 r^2 \quad (\bar{s}) = \frac{D_n \bar{s}}{\sqrt{1-B}} \quad (6)$$

$$\Gamma(\bar{s}) = \frac{D_n \bar{s}}{\sqrt{1-B}} \quad (7)$$

$$Z(\bar{s}) = \frac{Z_0}{\sqrt{1-B}} \quad (8)$$

$$B(\bar{s}) = \frac{2J_1\left(j\sqrt{\bar{s}}\right)}{j\sqrt{\bar{s}}\,J_0(j\sqrt{\bar{s}})} \quad (9)$$

where:

L = line length
ν_0 = mean kinematic viscosity
ρ_0 = mean fluid density
r_0 = radius of tube
D_n = dissipation number
J_0, J_1 = first and second order Bessel functions
K_B = Bulk modulus of the liquid
$Z_0 = \rho_0 c_0 / \pi r_0^2$ impedance constant
$c_0 = \sqrt{\frac{K_B}{\rho}}$ = fluid sonic speed
$\bar{s} = \frac{r^2}{\nu_0} s$ = normalized Laplace variable

Causality

If one thinks of the vector of variables on the right side of (5) as the input and the vector on the left as the output, the equation does not represent a causal physical system. Pressure and flow at the same location cannot be independently dictated at the same time. Algebraic manipulation can yield four alternative matirx equations in

causal form with seven distinct transfer functions. Equations 10-13 are the four possible causal conditions that can be formed from Equation 5[6].

$$\begin{bmatrix} P_a(s) \\ P_b(s) \end{bmatrix} = \begin{bmatrix} \frac{Z(s)\cosh(\Gamma(s))}{\sinh(\Gamma(s))} & -\frac{Z(s)}{\sinh(\Gamma(s))} \\ \frac{Z(s)}{\sinh(\Gamma(s))} & -\frac{Z(s)\cosh(\Gamma(s))}{\sinh(\Gamma(s))} \end{bmatrix} \begin{bmatrix} Q_a \\ Q_b \end{bmatrix} \tag{10}$$

$$\begin{bmatrix} P_a(s) \\ Q_b(s) \end{bmatrix} = \begin{bmatrix} \frac{Z(s)\sinh(\Gamma(s))}{\cosh(\Gamma(s))} & \frac{1}{\cosh(\Gamma(s))} \\ \frac{1}{\cosh(\Gamma(s))} & -\frac{\sinh(\Gamma(s))}{Z(s)\cosh(\Gamma(s))} \end{bmatrix} \begin{bmatrix} Q_a \\ P_b \end{bmatrix} \tag{11}$$

$$\begin{bmatrix} Q_a(s) \\ P_b(s) \end{bmatrix} = \begin{bmatrix} \frac{\sinh(\Gamma(s))}{Z(s)\cosh(\Gamma(s))} & \frac{1}{\cosh(\Gamma(s))} \\ \frac{1}{\cosh(\Gamma(s))} & -\frac{Z(s)\sinh(\Gamma(s))}{\cosh(\Gamma(s))} \end{bmatrix} \begin{bmatrix} P_a \\ Q_b \end{bmatrix} \tag{12}$$

$$\begin{bmatrix} Q_a(s) \\ Q_b(s) \end{bmatrix} = \begin{bmatrix} \frac{\cosh(\Gamma(s))}{Z(s)\sinh(\Gamma(s))} & -\frac{1}{Z(s)\sinh(\Gamma(s))} \\ \frac{1}{Z(s)\sinh(\Gamma(s))} & -\frac{\cosh(\Gamma(s))}{Z(s)\sinh(\Gamma(s))} \end{bmatrix} \begin{bmatrix} P_a \\ P_b \end{bmatrix} \tag{13}$$

Conversion to the Time Domain

The models derived above can be used in several different ways. For our current purposes we wish to convert the component model to a finite set of state space equations. This can be done by first expressing the transcendental transfer functions in partial fraction expansion as a truncated infinite series of first or second order transfer functions, each of which can directly be converted to a differential equation.

The first step in a partial fraction expansion is determination of the poles, followed by the determination of the residues which constitute the numerator of each first order partial fraction term. In particular when the poles are found numerically, the determination of the residues require some unusual approaches that will be now discussed.

Finding the Residues

In order to determine the residues r_i corresponding to the poles(p_i) of a transfer function, consider the partial fraction expansion of the a transfer function *F(s)* represented as a ratio of polynomials in Equation 14:

$$F(s) = \frac{n(s)}{d(s)} = \frac{(s-z_1)(s-z_2)\ldots(s-z_{n-1})}{(s-p_1)(s-p_2)\ldots(z-p_n)} = \frac{r_1}{s-p_i} + \frac{r_2}{s-p_2} + \ldots + \frac{r_n}{s-p_n} \tag{14}$$

By multiplying Equation 14 by $s-p_i$ and finding the limit as s approaches p_i, the residue r_i can be represented as in Equation 15.

$$\lim_{s \to p_i} F(s)(s-p_i) = \lim_{s \to p_i} \left[\frac{r_1(s-p_i)}{s-p_1} + \frac{r_2(s-p_i)}{s-p_2} + \ldots + \frac{r_n(s-p_i)}{s-p_n} \right] = r_i \tag{15}$$

The term on the left side of Equation (15) is indeterminate at the poles and cannot be calculated directly since *F(s)* and $s-p_i$ equal ∞ and *0*, respectively, at $s=p_i$. When F(s) is given as a ratio of polynomials this is simply resolved by cancellation. This is not the case with the transcendental transfer functions here. As an alternative approach, the denominator can be represented by a Taylor series expansion around the poles.

$$d(s) = d(p_i) + d'(p_i)\frac{s-p_i}{1!} + d''(p_i)\frac{(s-p_i)^2}{2!} + \ldots \tag{16}$$

Simplifying this equation with the identity that $d(p_i)=0$, and dividing the equation by $s-p_i$ produces Equation 17.

$$\frac{d(s)}{s-p_i} = d'(p_i) + d''(p_i)\frac{s-p_i}{2!} + d'''(p_i)\frac{(s-p_i)^2}{3!} + \ldots \tag{17}$$

Now, by taking the limit of Equation 17 as s approaches p_i, the value of $d(p_i)/(s-p_i)$ is obtained as:

$$\lim_{s \to p_i} \frac{d(s)}{s-p_i} = \lim_{s \to p_i} d'(p_i) = d'(p_i) \tag{18}$$

Equation 19 is the derivative of the reciprocal of $F(s)$, simplified with the identity $d(p_i)=0$.

$$\left.\frac{d}{ds}\left[\frac{1}{F(s)}\right]\right|_{s=p_i} = \left.\frac{d}{ds}\left[\frac{d(s)}{n(s)}\right]\right|_{s=p_i} = \left.\frac{d'(s)n(s)-d(s)n'(s)}{(n(s))^2}\right|_{s=p_i} = \frac{d'(p_i)}{n(p_i)} \tag{19}$$

By substituting the reciprocal of Equation 18 into Equation 15, then applying the reciprocal of Equation 19, produces Equation 20.

$$\begin{aligned} r_i &= F(s)(s-p_i)\big|_{s=p_i} = \frac{n(s)}{d(s)}(s-p_i)\big|_{s=p_i} = \left[n(s)\frac{s-p_i}{d(s)}\right]_{s=p_i} \\ &= n(s)\left.\frac{1}{d'(s)}\right|_{s=p_i} = \left.\frac{n(s)}{d'(s)}\right|_{s=p_i} = \left.\frac{1}{\frac{d}{ds}\left[\frac{1}{F(s)}\right]}\right|_{s=p_i} \end{aligned} \tag{20}$$

This technique is useful when the transfer function can be calculated but is not in the form of a polynomial ratio. This method avoids the indeterminate condition as in Equation 15. For the hyperbolic transfer functions with Bessel function ratios, the values of the transfer function and its derivatives are easily obtained.

Time Domain Combination-Analytical Conversion (TDC-AC)

This will approach approximations at several steps as detailed in [6]. One approximation is the requirement for low viscosity. Evaluation of the Bessel functions and their ratio B is one example. B can be expressed as the ratio of two infinite products. The infinite product can then be substituted into the expressions for Z(s) and Γ(s) reduced to a product and ultimately an expression of the form

$$\frac{1}{\cosh(\Gamma)} = \sum_{i=1}^{k}\left[\frac{r_i}{(\bar{s}+p_i)}+\frac{r_i^*}{(\bar{s}+p_i^*)}\right] = \sum_{i=1}^{k}\left[\frac{a_i\bar{s}+b_i}{(\bar{s}^2+2\zeta_i\omega_i\bar{s}+\omega_i^2)}\right] \tag{21}$$

produces the complex poles of the transfer functions in (10)-(13). A similar process is required for 1/ sinh. Tabulated values [5] allow one to readily construct all of the transfer functions in those equations. From the simple transfer functions above a state space model can be constructed in one of several canonical forms.

Time Domain Combination-Numerical Conversion (TDC-NC)

The assumptions and approximations necessary to follow the TDC-AC may lead to questions of accuracy. Instead, the TDC-NC approach numerically searches for the values of the poles of the transfer functions in the component equations. Effectively a search is conducted for the zeros of cosh(Γ(s)) and sinh(Γ(s)), which are the denominators of the seven unique transfer functions. The MATLAB function fmins is used. Fmins employs a Nelder-Meads simplex algorithm.

It is necessary to produce suitable starting points for the numerical search. Consider the transfer function $1/\cosh(\Gamma(s))$. The relationship of cosine to its hyperbolic counterpart is: $\cos(x)=\cosh\left(x\sqrt{-1}\right)$. Also, it is known that the zeros of *cos(x)* occur at $x=n\pi/2$ where $n=1,3,\ldots$ and the value $\Gamma(s)=\frac{D_n s}{\sqrt{1-B}}\approx D_n s$ for low damping. Thus, the imaginary component of the pole can be approximated by:

$$\mathrm{Im}[p_{n,guess}]=\frac{2n-1}{2D_n}\pi i \tag{22}$$

The real part of the pole is approximated using a much more empirical formula:

$$\mathrm{Re}[p_{n,guess}]=\begin{cases}\dfrac{-1}{100D_n} & D_n \le 1\cdot10^{-7}\\ \dfrac{1}{\sqrt{D_n}} & D_n > 1\cdot10^{-7}\end{cases} \tag{23}$$

The purpose of Equations 22 & 23 is to provide guesses for the initial, low frequency poles. After the first pole is found, the subsequent guesses are extrapolated from the previous poles. The poles are assumed to be linearly spaced in the complex plane.

It is interesting to note that the Bessel functions become very large with increasing values of the imaginary portion of the argument argument. For instance $|J_1(100j)|=1.1\cdot10^{42}$. For *1000j*, the value becomes too large to be represented by MatLab's IEEE *long* format ($>10^{308}$). Fortunately, MatLab has a convenient way to avoid this

problem. The function besselj has an optional argument that causes the answer to be divided by $e^{\mathrm{Im(arg)}}$. This factor is canceled when the ratio of the Bessel functions is calculated. Thus, the ratio is calculated accurately without the necessity to represent ex remely large numbers.
The residues are again found using the method outlined in the section above. To use this method, the derivative of the transfer function must be calculated. For example, consider the transfer function $F(s)=\dfrac{\sinh(\Gamma(\bar{s}))}{Z(\bar{s})\cosh(\Gamma(\bar{s}))}$.
Equation 24 shows the manipulation of *F(s)*, with the identity that *cosh(Γ (p$_i$)=0.*

$$\left.\frac{n(\bar{s})}{d'(\bar{s})}\right|_{s=p_i}=\frac{\sinh(\Gamma(\bar{s}))}{\dfrac{d}{ds}\left[Z(\bar{s})\cosh(\Gamma(\bar{s}))\right]}=\frac{\sin(\Gamma(\bar{s}))}{Z(\bar{s})\Gamma'(\bar{s})\sinh(\Gamma(\bar{s}))}=\frac{1}{Z(\bar{s})\Gamma'(\bar{s})} \tag{24}$$

Equation 25 contains the derivative of $\Gamma(\bar{s})$, which is found to be:

$$\Gamma'(\bar{s})=\frac{d}{d\bar{s}}\frac{D_n\bar{s}}{\sqrt{1-B}}=D_n\frac{\sqrt{1-B}-\dfrac{\bar{s}}{2}(1-B)^{1/2}(-B')}{1-B}=\frac{1-B+\dfrac{B'\bar{s}}{2}}{(1-B)^{3/2}} \tag{25}$$

Equation 25 contains the derivative of the bessel function ratio *B*. The identities $J'_0(x)=-J_1(x)$ and $J'_0(x)=J_0(x)-J_1(x)/x$ and the substitution $x=j\sqrt{\bar{s}}\Rightarrow x'=\dfrac{1}{2\sqrt{\bar{s}}}$ are used to form the derivative of *B* in Equation 26.

$$B'=\frac{d}{d\bar{s}}\left[\frac{2J_1(x)}{xJ_0(x)}\right]=2\frac{J_1'(x)x'xJ_0(x)-J_1(x)(x'J_0(x)+xJ_0'(x)x')}{x^2J_0^2(x)}= \tag{26}$$

$$=B'=\frac{1}{2\bar{s}}\left[2-2B-\frac{\bar{s}}{2}B^2\right] \tag{27}$$

For transfer functions which contain poles located at the origin, the problem of determining the residue is complicated. Consider the transfer function $F(\bar{s})=\dfrac{Z(\bar{s})}{\sinh(\Gamma(\bar{s}))}$ which has a pole located at the origin. The residues are determined with Equation 27.

$$\left.\frac{1}{\dfrac{d}{d\bar{s}}\left[\dfrac{1}{F(\bar{s})}\right]}\right|_{s=p_i}=\left.\frac{Z(\bar{s})}{\Gamma'(\bar{s})\cosh(\Gamma(\bar{s}))}\right|_{s=p_i} \tag{28}$$

Evaluation of some of the parameters of Equation 27 are relatively simple for $\bar{s}=0$, as shown in Equation 28. In each case these values had to be obtained using L'Hopital's rule to evaluate the limit of the equation

$$B(0)=1 \qquad B'(0)=-\frac{1}{8} \qquad \Gamma(0)=0$$
$$\Gamma'(0^+)=\infty \qquad \Gamma'(0^-)=-j\infty$$

Clearly, Equation 27 is indeterminate since $Z=1/\sqrt{1-B}$ is infinite and $\Gamma(\bar{s})$ is infinite. Calculation of the limit of Equation 27 is algebraically involved. Fortunately, the symbolic limit of the equation may be obtained using the Symbolic Toolbox within MatLab. The limit in this case evaluates to Z_0/D_n. All residues for zero and nonzero poles for all seven distinct transfer functions have been analytically expressed in terms of the poles that will be numerically found. [8]

Frequency Domain Combination-Numerical Conversion (FDC-NC)

The strategy in the TDC approaches above is to enable standard techniques to convert to the time domain for a given component. A limited number of component modes are retained and the resulting system equation will have order equal to the sum of the order of all components. The accurate representation of N_s modes for the system will require more than N_s in total. This is essentially because the component modes are not system modes but effectively basis functions that allow the spatial variation of the system variables to be represented. The strategy with the current FDC approach is to combine components in the frequency domain where possible, retaining the infinite order transfer function. The transfer matrix representation of components in (1), and (5) make the serial combination a simple matter of multiplying these matrices together. The resulting equation form

is not causal. If the equations are converted to causal form an explicit relation between inputs and outputs is obtained. Alternatively, the noncausal form can be the basis for conversion to the time domain as shown in [1].

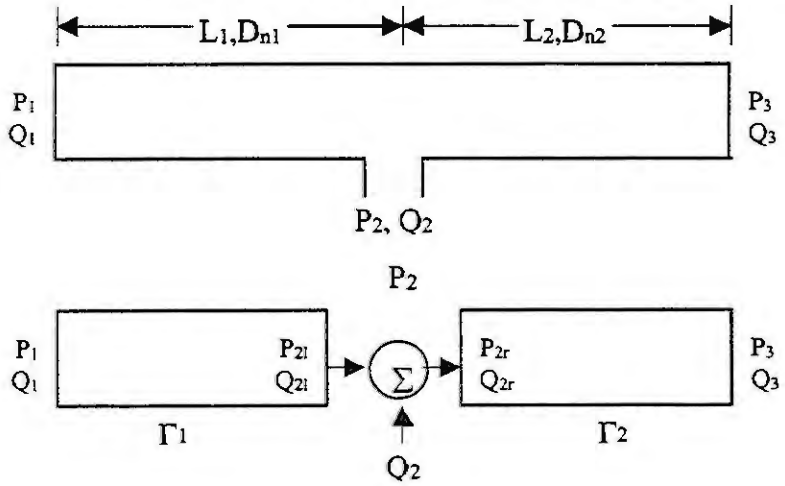
This paper has modified the previous work by incorporating new methods for solving for the residues. In the case of [1] it was necessary to use curve fitting techniques to find an appropriate residue, the numerator of the partial fraction expansions. Here the technique detailed above is much cleaner and more efficient. It does require analytical manipulation of the transfer functions using symbolic processing and that must be done for a given combination of components. Once completed however, the residues are explicitly known in terms of the values of the corresponding poles.

Numerical Example: Serial Combination of Fluid Lines

A simple but revealing example is to create a line model for two lines connected end to end, with an opening somewhere along the line as in Figure 1. The radii of the lines are equal and no restriction is made on the length of the lines. This selection allows comparison to known solutions in special cases. Combining the components and the conservation of flow and equality of pressure where the lines join produces a noncausal version of the model.

Equation 29 is not a realistic causality condition since both P_3 and Q_3 are specified. There are a number of ways this could be rearranged to produce a causal condition, for instance exchanging P_1 and P_3. In order to do this, consider the simple algebraic manipulation of the partitioned linear equation in Equation 30.

Figure 1. Two Fluid Line Element Combination



$$\begin{bmatrix} P_1(s) \\ Q_1(s) \\ P_2(s) \end{bmatrix} = \begin{bmatrix} \cosh(\Gamma_1+\Gamma_2) & -Z\sinh(\Gamma_1) & Z\sinh(\Gamma_1+\Gamma_2) \\ \frac{1}{Z}\sinh(\Gamma_1+\Gamma_2) & -\cosh(\Gamma_1) & \cosh(\Gamma_1+\Gamma_2) \\ \cosh(\Gamma_2) & 0 & \cosh(\Gamma_2) \end{bmatrix} \begin{bmatrix} P_3(s) \\ Q_2(s) \\ Q_3(s) \end{bmatrix} \tag{29}$$

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \Rightarrow$$
$$\begin{bmatrix} y_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} A_{11}^{-1} & -A_{11}^{-1}A_{12} \\ A_{21}A_{11}^{-1} & A_{22}-A_{21}A_{11}^{-1}A_{12} \end{bmatrix} \tag{30}$$

Applying the same manipulation from Equation 56 to Equation 55 produces the causal transfer function matrix in Equation 31.

$$\begin{bmatrix} P_3(s) \\ Q_1(s) \\ P_2(s) \end{bmatrix} = \begin{bmatrix} \frac{1}{\cosh(\Gamma_1+\Gamma_2)} & \frac{Z\sinh(\Gamma_1)}{\cosh(\Gamma_1+\Gamma_2)} & -Z\tanh(\Gamma_1+\Gamma_2) \\ \frac{\tanh(\Gamma_1+\Gamma_2)}{Z} & \frac{-\cosh(\Gamma_2)}{\cosh(\Gamma_1+\Gamma_2)} & \frac{1}{\cosh(\Gamma_1+\Gamma_2)} \\ \frac{\cosh(\Gamma_2)}{\cosh(\Gamma_1+\Gamma_2)} & \frac{Z\sinh(\Gamma_1)\cosh(\Gamma_2)}{\cosh(\Gamma_1+\Gamma_2)} & \frac{-Z\sinh(\Gamma_1)}{\cosh(\Gamma_1+\Gamma_2)} \end{bmatrix} \begin{bmatrix} P_1(s) \\ Q_2(s) \\ Q_3(s) \end{bmatrix} \tag{31}$$

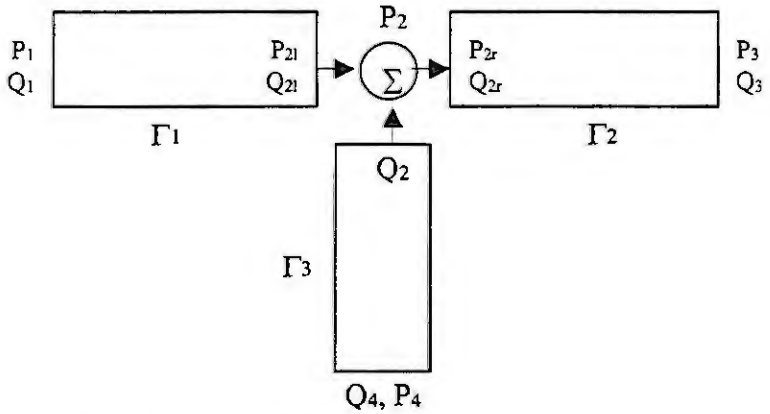
Comparing (29) and (31) exposes one of the advantages of the causal form. From (31) the poles of the serial combination are clearly the values of s the are the roots of $\cosh(\Gamma_1 + \Gamma_2)$. This is clear regardless of the specification of inputs or boundary conditions on the variables chosen as the input.

In each of the transfer functions of Equation 31, the combination $\Gamma_1+\Gamma_2$ appears. By setting the intermediate flow to zero, the system is reduced to the basic system of one line with two ends and extra information about the pressure within the flow element. Further, if P_2 is ignored, Equation 31 takes on the exact form of a single line, verifying the equations.

Branched System

A further generalization involves branches in the fluid line as shown in Figure 2.

Figure 2. Three Line Branch System



An example of a causal form is as follows:

$$\begin{bmatrix} Q_4(s) \\ P_3(s) \\ P_1(s) \end{bmatrix} = \frac{1}{K} \begin{bmatrix} -\sinh(\Gamma_2) & \cosh(\Gamma_1+\Gamma_2)\cosh(\Gamma_3)/Z + \cosh(\Gamma_1)\cosh(\Gamma_2)\cosh(\Gamma_3)/Z & -\dfrac{\cosh(\Gamma_1)}{Z} \\ \sinh(\Gamma_3) & \dfrac{\cosh(\Gamma_1)}{Z} & -\sinh(\Gamma_1+\Gamma_2)\sinh(\Gamma_3)/Z - \cosh(\Gamma_1)\cosh(\Gamma_2)\cosh(\Gamma_3)/Z \\ Z\sinh(\Gamma_1+\Gamma_2)\sinh(\Gamma_3)+ Z\sinh(\Gamma_1)\sinh(\Gamma_2)\cosh(\Gamma_3) & \sinh(\Gamma_2) & \sinh(\Gamma_3) \end{bmatrix} \begin{bmatrix} Q_1(s) \\ P_4(s) \\ P_3(s) \end{bmatrix} \quad (32)$$

where: $K = \cosh(\Gamma_1+\Gamma_2)\sinh(\Gamma_3) + \cosh(\Gamma_1)\sinh(\Gamma_2)\cosh(\Gamma_3)$.

Clearly, the roots of K are the poles of this system for any simple boundary conditions on the input variables. An analytical solution for these roots is not available but numerical searches for the roots is relatively straight forward. The techniques for finding the residues is again possible and is applied to the case of poles at the origin differently than the general case for poles elsewhere.

Choice of Time Domain Form

Two canonical forms are readily produced from a transfer function representation of second order dynamics: controllable form and observable form. In controllable form numerical problems can result due to poles at the origin. The steady state value of flow, for example, will depend on the difference between the integral of pressure at each of two inputs. Control canonical form integrates each pressure first then combines the result with a difference. As time gets large the two integrals must both become large and have a difference of zero to represent zero steady state flow. Observer canonical form combines the pressures prior to integration then integrates. Hence the numerical problems associated with a small difference of large numbers never arises.

Numerical Results and Comparison

The serial combination of lines has been simulated to give representative results to refer to when discussing the advantages of the three techniques described above. Properties were approximately those that would be found in automotive components such as fuel lines. The parameters used were are in the Table below.

Table 1: Parameters for Serial Line Combination

L_1	L_2	r_0	c_0	ν_0	ρ_0
15 cm	22.5 cm	6.35 mm	2848 m/s	$3.52\text{x}10^{-7}$cm	680 kg/m^3

The system will be subjected to a step pressure response at the open end of a line blocked at the other end. The result of applying a sudden pressure change at the end of a line is a pressure wave. This wave travels down the length of the line, and is reflected at the opposite end of the line, which is blocked. The nature of this wave is "square". With no friction, the wave would continue to reflect from end to end with this square shape. The dissipation used in our model causes the oscillations to decay with time. This decay slowly erodes the wave shape until no oscillations remain. At steady state, the pressure at the blocked end will be equal to the pressure at the open end, and the flow at the open end will be zero. This experiment is similar to the classic "waterhammer" example in which there is an initial steady non-zero flow which is abruptly stopped, for example with a valve. The simple nature of the exact response allows one to readily compare the simulations to reality.

Figures 3 and 4 show the response of Q_1 with Q_2 set to zero and a unit step applied to P_1. Figure 3 shows the initial response. All of the traces exhibit the same general response, with the ripple in the theoretically flat portions of the time response having higher amplitudes for Methods 1 (TDC-AC_) and 2 (TDC-NC) than for Method 3 (FDC-NC). The vertical dotted lines represent the wave travel time calculated simply from $T=2L/c_o$.

Figure 13 shows the response after 20 oscillations in the line. Both time domain combinations, Methods 1 and 2 exhibit the decay of the ``square" nature of the single element response. Additionally, the frequency of the reflection of the square wave is different from the predicted value in each of the two element models. The phase of the square wave response is deceptive from Figure 13, because after approximately 14 oscillations, Method 1 and Method 2 were 180^0 out of phase, with Method 2 more closely corresponding to the Method 3 model.

Conclusions

This paper has presented ways in which to model fluid line elements. The method of combination of individual models in the frequency domain before transformation into the time domain seems to provide improved performance of the model because of the ability to accurately represent modes of combined systems, instead of relying on the component modes of subsystems. For this reason, fewer modes are necessary to represent the behavior of the combined system. In the case considering branched lines with radically different lengths (not shown in this paper), the frequency domain combination model poorly represented short line dynamics.

Figure 3. Two Line Blocked Comparison, Initial Response

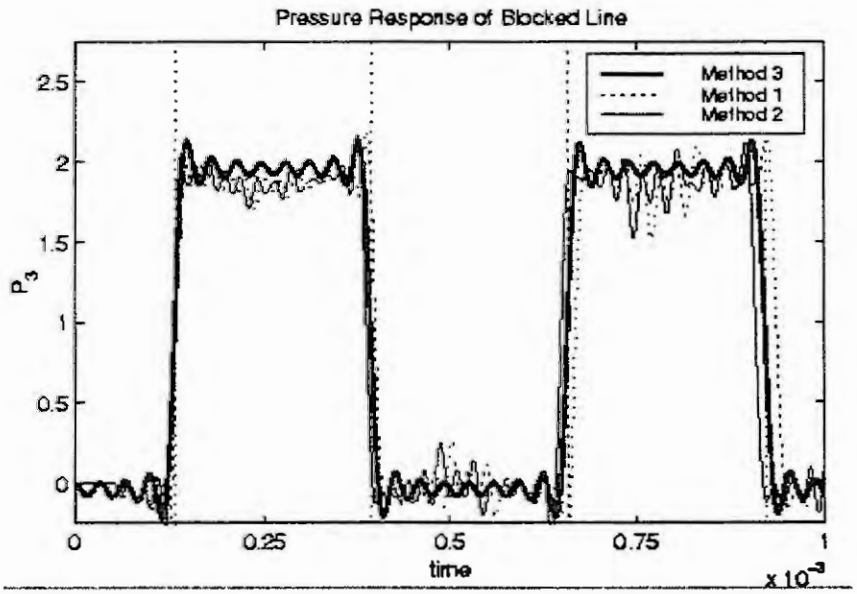
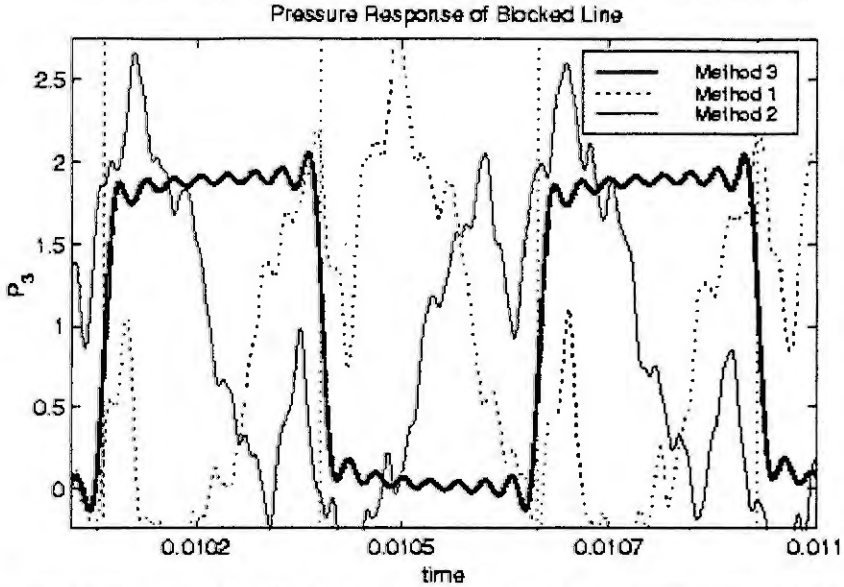


Figure 4. Two Line Blocked Comparison, Response After Twenty Cycles



This paper has demonstrated that the combination of relatively few components in the time domain reduces the accuracy of the system response. Error is accumulated since the system is constructed of many subcomponents which are individually approximated, providing many sources for error. These errors can consist of numerical dispersion in the system, or artificially high amplitude, high frequency components. By increasing the number of modes in the subsystems, the errors can be reduced, but this requires the model to become increasingly large. Frequency domain combination attacks these problems at the source, component modes. For every frequency domain combination, one more set of component modes is eliminated. The case of widely differing line lengths presented a case where the construction of such a frequency domain model has practical difficulties. Thus, to construct a system, a hybrid approach must be used to form a complete system. This method would use frequency domain combination where practical, including many modes to improve the interaction with other subsystems. Additional details are found in [8].

Acknowledgements

Partial funding for this research was provided by a grant from Ford Motor Company.

References

1. Book, W. J. and Majette, M. Controller design for flexible, distributed parameter mechanical arms via combined state space and frequency domain techniques. *Journal of Dynamic Systems, Measurement and Control*, Vol 105, pp 245-254, December 1983.
2. Goodson, R. E. and Leonard, R.G. A survey of modeling techniques for fluid line transients. *Journal of Basic Engineering*, Transaction of ASME, 94, June 1972. Series D.
3. Glidwell, J. M., Yang, W.C. and Chuo, G. K. An on-board diagnostic strategy for multi-port electronic fuel injection systems using fuel transient analysis. ASME, Advanced Automotive Technologies, Vol 52, pp 257-265, 1993.
4. Heally, A. J. and Hullender, D. A. State variable representation of modeal approximations for fluid transmission line systems. *Journal of Dynamic Systems, Measurement and Control*, ASME, November 1981. Special Symposium on Volume of Fluid Transmission Line Dynamics.
5. Hsue, C. Y-Y and Hullender, D. A. Modal approximations for the fluid dynamics of hydraulic and pneumatic transmission liens. *Fluid Transmission Line Dynamics*, II, November 1983. ASME Special Publication, New York.
6. Hullender, D. A. and Heally, A. J. Rational polynomial approximations for fluid transmission line models. *Fluid Transmission Line Dynamics*, November 1981. Special Publication of the ASME Annual Winter Meeting, Washington, D.C.
7. Iberall, A. S.. Attenuation of oscillatory pressures in instrument lines. *Journal of Research*, 45 (R.P. 2115), July 1950. National Bureau of Standards.
8. Watson, C., "Modeling of Pressure Transients in Fuel Injection Lines," M.S. Thesis, School of Mechanical Engineering, Georgia Institute of Technology, Dec, 1999.
9. Yang, W. C. and Tobler, W.E. Dissipative modal approximation of fluid transmission lines using linear friction model. *Journal of Dynamic Systems, Measurement and Control*, 113, March 1991.

A Fast Integration Algorithm for Three-Way Catalytic Converters PDE Models

Luigi Glielmo and Stefania Santini
Dipartimento di Informatica e Sistemistica, Università di Napoli Federico II,
via Claudio 21, 80125 Napoli, ITALY; e-mail: {glielmo,stsantin}@unina.it

Abstract

We present a PDE model of three-way catalytic converters used on commercial gasoline-powered vehicles. In order to speed-up simulation times, we developed a fast integration algorithm based partly on a 'method of lines' space-discretization, partly on the 'method of characteristics' for 'quasi linear' hyperbolic PDEs, the separation being allowed by a two time scale analysis of the system.

1 Introduction

Nowadays all gasoline-powered cars produced have to be equipped with the so-called three-way catalytic converters (TWC). The aim of this device is to transform the pollutant gases coming out of the combustion process, namely carbon oxide, hydrocarbons and nitrogen oxide, into less dangerous chemical species such as carbon dioxide (obtained through oxidation of carbon oxide) or nitrogen (obtained through reduction of nitrogen oxide). The catalitic components present on the active substrate of TWC's help those reaction to occur fast enough so as to guarantee a decrease of pollutant concentrations exceeding 95%. In the first years after introduction of TWC, designers were more interested to the steady-state behavior of the device, when its temperature reaches about 600°K and composition of air/fuel mixture in the combustion chamber can be well regulated around its stoichiometric ratio, where TWC oxidation and reduction efficiencies are both satisfactory. Recently the analysis and control of TWC transient behavior has received more attention because (*i*) the emission of pollutant gases during the warm-up phase is very significant, in relative terms, with respect to the overall production during an average driving cycle; and (*ii*) the new generation TWCs, to be used with DISC (direct injection stratified charge) engines, reach maximum efficiency working with a nonconstant air/fuel ratio.

Since our final goal is the development of optimization and control algorithm for TWC to be possibly employed in the design of overall real-time engine control strategies, we concentrated our efforts on reduced order TWC models, trying to include only the relevant features of the dynamic behavior. The model we present here, discussed in detail in [2], has been obtained by assuming that the adsorption coefficient between gas and substrate is infinite; this idealization means that the adsorption phenomenon is infinitely faster than the chemical reaction taking place on the substrate, and is a reasonable simplification during the warm-up phase.

This reduced order model is still a distributed parameter model and its simulation times would still be prohibitive without using a special care in the design of the integration algorithm. We based our algorithm on the fact that (*i*) the working of a TWC derives from the interplay of thermal phenomena (thermal exchanges between gas and substrate and thermal energy generated by chemical reactions) and chemical reactions on the substrate; (*ii*) the thermal phenomena are much slower than the chemical phenomena. Consider now that, when dealing with two-time-scale lumped parameter systems, described via singularly perturbed ODEs (*e.g.*, [6]), one computes the "slow" subsystem by replacing the fast dynamics with algebraic relations. Here, similarly, we suggest to replace the chemical part of the equation with simple algebraic relations which summarize the conversion efficiencies of the TWC; on the other side, since this efficiencies depend on the temperature, their values have to been modified from time to time by integrating the appropriate chemical equations. In this way we managed to decrease the simulation time of an entire driving cycle from 20 minutes to 10 seconds!

2 A Dynamic Model of TWC

In this section we present the dynamic TWC model described in detail in [1, 2]. It is a monodimensional PDE (partial differential equation) model where the non-uniform flow distribution at the monolith face is neglected:

$$v_m(t,x)\,\rho_g(t,x)\,c_g\frac{\partial T_g}{\partial x} + hG_A(T_g - T_s) = 0, \tag{1a}$$

$$(1-\epsilon)\rho_s c_s\frac{\partial T_s}{\partial t} = (1-\epsilon)\lambda_s\frac{\partial^2 T_s}{\partial x^2} + hG_A(T_g - T_s) +$$

$$- h_{ext} S_{ext}(T_s - T_{ext}) - \Delta H^T R'(X, T_s), \tag{1b}$$

$$(1-\epsilon)\frac{\partial X}{\partial t} = -v_m(t,x)\frac{\partial X}{\partial x} - R(X, T_s). \tag{1c}$$

We consider p chemical species participating to q catalytic reactions and the above equations describe the energy and the mass equilibrium. Pedices 'g' and 's' stand for gas and substrate (the reactive surface); T is temperature; X is the p-vector of species concentrations expressed in [mol/m^3] units and assumed equal in the gas and solid phase in view of the infinite-adsorption hypothesis (see [1, 2] for more details); $K_D = \mathrm{diag}(k_{D,1}, \ldots, k_{D,p})$ is the diagonal matrix of adsorption coefficients between the gas phase and the substrate for the various species; R is the p-vector of specific reaction rates for the species and R' is the q-vector of specific reaction rates for the chemical reactions, both depending on substrate temperature and concentrations; ΔH^T is the q-vector of the heat produced by the catalytic reactions; the independent variables t and x are respectively the time and the axial position along the monolith; the various other coefficients are illustrated in the Appendix.

The chemical model adopted includes six chemical species (CO, C_3H_6, CH_4, NO, H_2, O_2; hence $p = 6$); they take part into five chemical reactions ($q = 5$) that include the oxidation of CO, H_2, HC, as well as the NO reduction:

$$\begin{array}{cc} CO + 0.5O_2 \rightarrow CO_2, & C_3H_6 + 4.5O_2 \rightarrow 3CO_2 + 3H_2O, \\ CH_4 + 2O_2 \rightarrow CO_2 + 2H_2O, & CO + NO \rightarrow CO_2 + 0.5N_2, \\ H_2 + 0.5O_2 \rightarrow H_2O. & \end{array}$$

The boundary and initial conditions are ($t \geq 0, x \in [0, L]$)

$$\frac{\partial T_s}{\partial t}(t, L) = 0 \quad \text{(Adiabatic constraint)}, \quad T_g(t, 0) = T_g^*(t),$$
$$X(t, 0) = X_g^*(t), \quad T_s(0, x) = T_s^*(x), \quad X(0, x) = X^*(x),$$

where $T_g^*(t)$ and $X_g^*(t) = (X_{g,1}^*(t), \ldots, X_{g,6}^*(t))^T$ are respectively the temperature of the exhaust gas and the concentrations of the chemical species at the inlet of the TWC, $T_s^*(x)$ is the initial temperature of the substrate, $X^*(x) = (X_1^*(x), \ldots, X_6^*(x))^T$ are the initial concentrations and L is the TWC length.

3 Integration Algorithm

Before developing the algorithm of numerical integration, the set of equations (1) has been cast into a dimensionless form; from now on, for the sake of simplicity, the same symbols will refer to dimensionless quantities.

Solving Equation (1c)

For reader's convenience, we rewrite here the concentration equations (1c) as follows

$$\frac{\partial X(t,x)}{\partial t} + \frac{u_m(t,x)}{(1-\epsilon)}\frac{\partial X(t,x)}{\partial x} = -\frac{R(X(t,x), T_s(t,x))}{(1-\epsilon)}. \tag{3}$$

The initial conditions are prescribed along the axes of the (t,x)-plane $X(t,0) = X_g^*(t)$ $t \geq 0$, $X(0,x) = X^*(x)$ $x \in [0,1]$. *If the temperature pattern $T_s = T_s(t,x)$ were known*, (3) would be a system of 'quasi linear' hyperbolic PDEs and could be solved using the characteristics method [4, 5] yielding the ODEs

$$\frac{dt}{ds} = 1, \qquad \frac{dx}{ds} = \frac{u_m(t(s), x(s))}{(1-\epsilon)}, \qquad \frac{dX}{ds} = -\frac{\widetilde{R}(t(s), x(s), X(s))}{(1-\epsilon)}, \tag{4}$$

where $\widetilde{R}(t, x, X) \triangleq R(X, T_s(t, x))$.

In particular the first equation in (4) relates the parameter s of each characteristic curve to the time t; the second equation (4) describes the motion of the particle along the characteristic curve from $x = 0$ (inlet of TWC) to $x = 1$ (outlet of TWC); the third equation (4) describes the change of concentrations as the particle moves along the characteristic and, hence, along the converter. It is interesting to notice that the characteristics method allows a 'Lagrangian' approach to the problem since it describes the motion and the effects of chemical reactions on the single gas particle.

Since u_m is always positive, $x(s)$ is invertible in view of (4) and we can define new variables $\widetilde{X}(x)$, $\widetilde{t}(x)$ as $\widetilde{X}(x) \triangleq X(s(x))$, $\widetilde{t}(x) \triangleq t(s(x))$. Thus, along a characteristic curve, system (4) simply reduces to

$$\frac{d\widetilde{t}}{dx} = \frac{(1-\epsilon)}{u_m(\widetilde{t}, x)}, \qquad \frac{d\widetilde{X}}{dx} = -\frac{\widetilde{R}(\widetilde{t}(x), x, \widetilde{X}(x))}{u_m(\widetilde{t}(x), x)}, \tag{5}$$

where $x \in [0,1]$ and the initial conditions are prescribed as $\widetilde{t}(0) = \hat{t}$, $\widetilde{X}(0) = X_g^*(\hat{t})$.

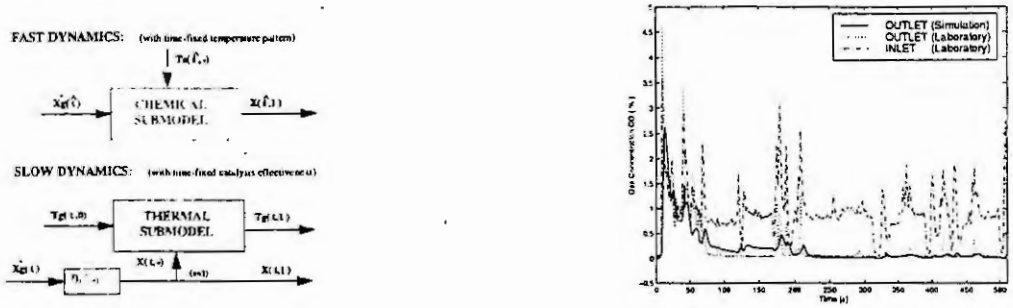


Figure 1: First frame: Approximate system in the time interval $[\hat{t}, \hat{t} + \Delta t]$. Second frame: CO time history.

Numerical Integration

Actually equation (1c) in the set of PDEs (1) cannot be solved as above because the function $T_s(t, x)$ is not *a priori* known but it is a solution of (1a)–(1b). Conversely, the concentrations X in equation (1b) are solution of (1c). In other words, the system (1) is coupled. However we can by-pass this difficulty by noticing that, in practice, the thermal dynamics described by (1a)–(1b) are much slower than the chemical dynamics of (1c). In particular the time spent by each gas particle inside the converter is very short compared to the time-scale of the thermal phenomenon.

It is thus possible to set up an approximate integration scheme (see figure 1, first frame) inspired to the a singular perturbation approach (*e.g.*, see [7, 6]):

- Consider the first equation in (5) and notice that the time spent by a gas particle to cross the converter, *i.e.* $\widetilde{t}(1) - \widetilde{t}(0)$, is small compared to the characteristic time of thermal phenomena. This enables us to set $\widetilde{t}(x) = \hat{t}$ for all $x \in [0, 1]$ and the second equation in (5) becomes

$$\frac{d\widetilde{X}}{dx} = -\frac{\widetilde{R}(\hat{t}, x, \widetilde{X}(x))}{u_m(\hat{t}, x)} = -\frac{R(\widetilde{X}(x), T_s(\hat{t}, x))}{u_m(\hat{t}, x)}, \tag{6}$$

  In other words we integrate (1c) separately from (1a)–(1b) by considering the temperature T_s to depend only on the space variable x. The solution of equation (6) depends on the initial conditions $\hat{t}$, $X_g^*(\hat{t})$ and the temperature pattern $T_s(\hat{t}, \cdot)$. We will simply indicate this by writing $\widetilde{X}(x; \hat{t})$.

- Now, to solve the thermal equations (1a)–(1b), we define the effectiveness of catalysis along the TWC as

$$\eta_i(\hat{t}, x) := \frac{\widetilde{X_i}(x; \hat{t})}{X^*_{g,i}(\hat{t})}, \qquad i = 1, \ldots, 6; \tag{7}$$

  then, on the time interval $[\hat{t}, \hat{t} + \Delta t]$, we use the linear approximation $X_i(t, x) \approx \eta_i(\hat{t}, x) X^*_{g,i}(t)$, $i = 1, \ldots, 6$. The width Δt must be chosen so that for $t \in [\hat{t}, \hat{t} + \Delta t]$

$$\Delta T(t) := \max_{x \in [0,1]} |T_s(t, x) - T_s(\hat{t}, x)| \leq \Delta_T, \quad \Delta X_i(t) := |X^*_{g,i}(t) - X^*_{g,i}(\hat{t})| \leq \Delta_{X,i} \quad i = 1, \ldots, 6, \tag{8}$$

  where Δ_T and $\Delta_{X,i}$, $i = 1, \ldots, 6$, are, respectively, fixed temperature and concentrations thresholds.

Equations (1a)–(1b) have been solved using a finite difference scheme ('method of lines') [8]. The distributed parameter model is converted into a lumped one by a finite difference scheme, thus considering a discrete number of spatial elements ('slices' of the converter), each described by time-varying variables. By this way from the temperature PDEs a system of ODEs is obtained which can be solved through usual integration packages.

The outline of the integration algorithm on the interval $[0, t_{fin}]$ follows:

$t := 0$;
$T_s^*(x) := T_s(0, x)$; (initial pattern of substrate temperature)
while $t \leq t_{fin}$
  solve the chemical equations with a temperature pattern T_s^*;
  evaluate $\eta_i(x) \quad x \in [0, 1] \qquad i = 1, \ldots, 6$;
  $\Delta T := 0$;

$\Delta X_i := 0 \quad i = 1, \ldots, 6;$
while $(\Delta T \leq \Delta_T)$ and $(\Delta X_i \leq \Delta_{X,i} \quad for \quad i = 1, \ldots, 6)$ and $(t \leq t_{fin})$
$X_i(t, x) = \eta_i(x) X^*_{g,i}(t), \quad i = 1, \ldots, 6;$
solve the thermal equations;
$T^*_s(x) = T_s(t, x);$ (update the pattern of temperature)
evaluate ΔT;
evaluate $\Delta X_i \quad i = 1, \ldots, 6;$
increment t;
end while
end while

4 Conclusions

The algorithm has been developed on Matlab 5.2/Simulink 2.0 environment with the support of C compiled S-function to shorten the computational time and the simulation of 510 real seconds of the warm-up along an FTP cycle takes 10 sec on a PC, Intel Pentium II 350MHz Processor, 128 Mb RAM.

Figure 1 (second frame) show, as an example, the CO time history obtained by the model compared to experimental data along the transient thermal phase of a full legislated USA drive cycle, FTP cycle (Federal Test Procedure), for a gasoline-powered passenger vehicle. In the figure a comparison between real data and model output (both referred to the outlet of the TWC) is reported.

5 Appendix

c_g	J/kg K	specific heat capacity of gas	c_s	J/kg K	specific heat capacity of substrate
R_i	mol/m^{2}sec	specific reaction rate for species i	R'_l	mol/m^{2}sec	specific reaction rate for the chemical reaction l
G_A	m^2/m^3	active area/volume ratio of the monolith	h_{ext}	W/m^2 K	heat transfer coefficient
$k_{D,i}$	m/sec	mass transfer coefficient for species i	S_{ext}	m^2/m^3	external area/volume ratio
T_{ext}	K	external temperature	u_m	m/sec	mean gas velocity in monolith
ΔH_i	J/mol	heat of ith-reaction	ϵ		void fraction
ρ_g	kg/m^3	gas density	ρ_s	kg/m^3	substrate density
$X_{g,i}$	mol/m^3	gas phase concentration of i-th species	$X_{s,i}$	mol/m^3	solid phase concentration of i-th species
h	W/m^{2}K	convective heat transfer coefficient	λ_s	W/m K	substrate thermal conductivity

References

[1] Glielmo, L., S. Santini, G. Serra, "A Two-Time-Scale Infinite-Adsorption Model of Three Way Catalytic Converters", *Proc. American Control Conference*, San Diego, California, 1999, pp. 2683–2687.

[2] Glielmo, L., and S. Santini, "A Two-Time-Scale Infinite-Adsorption Model of Three Way Catalytic Converters during the Warm-up Phase", submitted for publication.

[3] Heywood J. B., *Internal Combustion Engine Fundamentals*, McGraw-Hill, New York, 1988.

[4] Jeffrey, A., *Quasilinear Hyperbolic Systems and Waves*, Pitman Publishing, London, 1976.

[5] John, F., *Partial Differential Equations*, Springer-Verlag, New York, 1975.

[6] Kokotović, P. V., H. K. Khalil and J. O'Reilly, *Singular Perturbation Methods in Control: Analysis and Design*, Academic Press, London, 1986.

[7] Lin, C. C., and L. A. Segel, *Mathematics Applied to Deterministic Problems in the Natural Sciences*, Society for Industrial & Applied Mathematics, 1988.

[8] Schiesser, W. E., *The Numerical Method of Lines: Integration of Partial Differential Equations*, Academic Press, San Diego, 1991.

AN OBJECT-ORIENTED DATA MODEL TO CAPTURE LUMPED AND DISTRIBUTED PARAMETER MODELS OF PHYSICAL SYSTEMS

Jörg Hackenberg, Claudia Krobb, Wolfgang Marquardt [1]
Lehrstuhl für Prozesstechnik der RWTH Aachen
Turmstraße 46, 52056 Aachen, Germany

Abstract. In this paper a conceptual data model to capture mathematical models of physical systems and some of the knowledge collected during the establishment of these models is sketched. The data model is divided into two parts, one capturing the mathematical aspects and the other the physical interpretation of the equations. A more detailed version of this paper will be available at *http://www.lfpt.rwth-aachen.de.*

1 Introduction

Data models are the foundation of software development. In the phase of problem analysis data models are used to capture the main concepts of the application domain. The conceptual data model introduced here will be used as the basis for the development of a chemical process modeling system as presented in [6]. The modeling system will consist of several individually implemented, interacting tools. Probably none of these programs will use all of the concepts presented in this data model. The main purpose of the conceptual data model is rather to provide a common understanding of the concepts involved in the formulation of models of physical systems. Most of the tools will use simplified versions of the data model.

Characteristics of models of physical systems. From a mathematical point of view models of physical systems span the range from algebraic to integro-partial-differential-algebraic equation systems. The type of the resulting equation system depends on the decision to model the stationary or dynamic behavior of the systems and the decision to consider system quantities as distributed in spatial or substantial coordinates.

The domains of the values and the ranges of the functions representing a process quantity in a process model can be continuous such as a temperature or discrete like the number of trays in a distillation column. Discrete changes in the structure of a system may appear, i.e. if in multi-phase systems a phase emerges or vanishes. Those changes in the physical system are reflected in the emergence, disappearance or change of equations in the equation system. This is similarly true for system quantities and the corresponding functions and values.

If we consider the physical interpretation of the mathematical formulas usually written down during model development, we will find different abstraction levels. There might be a general local three-dimensional balance involving general quantities like a density of an extensive quantity. This balance is used as a starting point for model development and will be specified and transformed to a specific balance like a local one-dimensional mass balance involving mass density quantities. A process of model development like this is described in [4].

Structure of the data model. The discussion of models of physical systems has shown that we have to deal with two domains during the development of those models. One domain is formal mathematics, the other domain involves the physical concepts to interpret the formulas.

Therefore the data model is divided into two partial data models which are related by association links. The *mathematical* partial data model describes the mathematical characteristics of the equations, while the other partial data model describes their physical interpretation. We will refer to it as the *physical* partial data model in this paper.

According to Bunge [3] mathematics can be seen as a language used to express certain characteristics of the physical concepts. It is pointed out that the language definition does include the symbols and syntax rules to define sentences as well as formal rules for their transformation. The latter are called the (formal) logic of the language. When we associate the symbols and sentences of the language with physical concepts, we add semantics to the formulas. Examples of such associations are identifying a symbol like T with a temperature and an equation as a balance.

[1] **Acknowledgements.** This work has been supported in part by the *European Union* (Brite-EuRam/IMS project No. 26691 Global CAPE-OPEN). Claudia Krobb acknowledges support by the *Deutsche Forschungsgemeinschaft* (Graduiertenkolleg "Informatik und Technik").

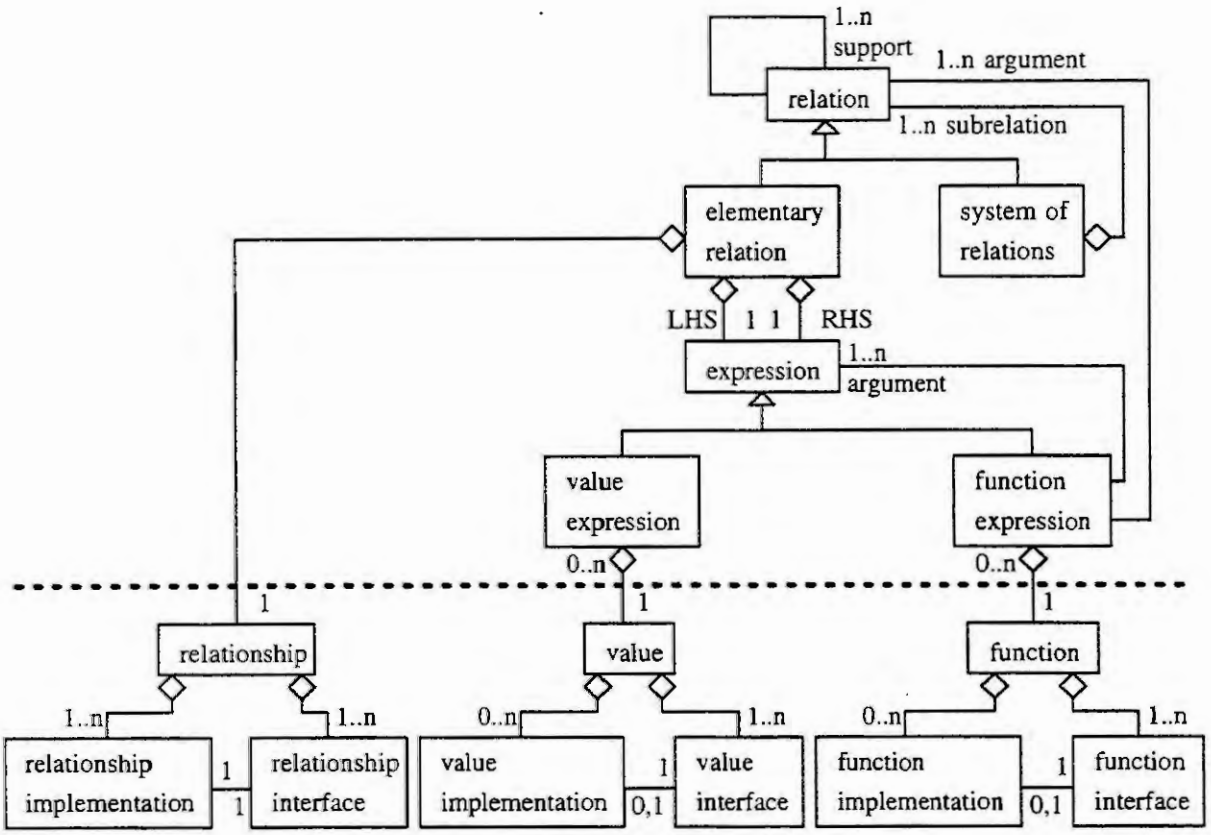


Figure 1: The mathematical partial data model.

One advantage of this strict differentiation between mathematics as the language and physical concepts for their interpretation is that e.g. authors of numerical codes do not have to deal with the interpretation of the mathematical objects. On the other hand the construction of design tools for (artificial) physical systems will mainly involve the *physical* partial data model. This is an advantage of this new data model compared to an earlier model proposed before [1].

The next section describes the *mathematical* partial data model. Thereafter we discuss the *physical* partial data model for the interpretation of the concepts of the *mathematical* partial data model. Due to the limited length of this paper both models will only be sketched, with an emphasis on the *mathematical* part.

2 The *mathematical* partial data model

The structure of the mathematical model is an extension of the definitions given in [2] translated into an object-oriented representation. An overview of the classes introduced is given in Figure 1. The model can be divided in two levels as indicated by the broken line. The part above the line describes the structure/syntax of the mathematical language, while the lower part is more involved in describing the logic of the language [3].

Relations. The central class of the data model is the class *elementary relation*. This class represents objects like equations, inequalities and so on. The class *relationship* describes the logical concepts of the relations like equality, but also relationships like is-element-of. This construct of separating concepts and their appearance in a formula facilitates multiple use of the same concept in different formulas.

Since we do not want to only represent single *elementary relations* we have introduced the superclass *relation*. This class can be either an *elementary relation* or a *system of relations*. The *system of relations* is aggregated from instances of the class *relation*. This mechanism allows us to build *systems of relations* with several levels of aggregation. All relations aggregated this way have to hold simultaneously.

Relations can be associated via the *support* to other *relations*. The support association represents

the $\forall$ in mathematical notation. These support *relations* must hold if the associated *relation* should be considered valid. For example, consider the set of all points of a volume. A *relation* can be defined to describe exactly these points. This *relation* can be used as a *support relation* for a local material balance which is valid at the points within the volume. The construct also gives us the means to describe the discrete changes of the model structure and composition as described in the introduction.

Expressions. The *elementary relations* comprise a left hand side and a right hand side *expression*. The *expressions* are divided into *function* and *value expressions*. The *function expressions* represent expressions like sums and products or, as an abstract example, $f(x)$. The *function expression* class is related to the *function* class which describes the formal mathematical mappings from domain sets to an image set. The argument of a *function expression* can be defined either by other *expressions*, like x in $f(x)$, or by *relations* which will be used, for example, to describe the integration domain of an integral operator which will mathematically be interpreted as a set itself and not an element of a set.

The *value expressions* represent *values* in the formulas. The *values* are elements of a set. They have terminal character for an *expression* since they do not have any arguments.

Functions, values and relationships. *Functions*, *values* and *relationships* are each aggregated from an *interface* and an *implementation* class. The *interface* classes will refer to the possible domain sets of the left hand side and right hand side *expression* in case of a *relationship*, to the domain and image sets in case of a *function* and to the domain set a *value* belongs to. On the other hand the *implementation* classes describe such things as a method to check a *relationship*, the actual object representing a *value* and a method to formally evaluate a *function*.

There can be multiple implementations and interfaces. This allows for the explicit description of an overloading mechanism, as used by many programming languages, e.g. C++. Overloading appears in our models if we consider an equation not only as relating numbers but also as relating physical dimensions and physical units.

Variables and constants. The mathematical constant concept is a representation of a particular element of a domain set. In case of a variable concept only the domain set is specified. No particular element of this set is associated with the variable concept. In the data model the concepts of variables and constants do not appear explicitly. These concepts are realized with the cardinality of the associations between *implementations*, *interfaces* and the aggregates thereof. A constant *value* will be represented by an *implementation* and *interface* object. Thus an *implementation* object representing the value of, e.g. π, and an *interface* object representing the set of real numbers $\mathbb{R}$ will appear in the data model to represent the constant *value* π. If x was represented as a variable *value* only an *interface* object would be present in the data model. In the same way variable and constant *functions* can be modeled. No variable *relationships* are known to appear in mathematical models of physical systems, thus a *relationship* must have an *implementation* and an *interface*. It is pointed out that if the *function interfaces* only include domain and image sets of a function the sets of functions a variable *function* represents can only be described roughly. If we want a more precise description of this set of functions we will need to add additional information to the *function interface*.

3 The partial data model for the *physical* interpretation concepts

The top level structure of the partial data model for the *physical* interpretation concepts is shown in Figure 2. At each class in the Figure a generalization/specification tree is indicated. These trees are not discussed in this paper. Some examples can be found in [5]. These taxonomies provide the opportunity to model different levels of abstraction as mentioned in the introduction.

System quantities. The *system quantities* are the quantitative attributes chosen to describe a physical system. This class includes concrete members such as temperatures, the number of trays in a distillation column, the length of a time interval or the state of a valve (open/close), as well as abstract members like extensive thermodynamic quantities or general fluxes. If the *system quantity* is distributed there will be an association with a *physical domain* defining the distribution domain.

The *system quantities* will be associated to objects of the *function* or *value* class from the mathematical model. Distributed quantities will be linked to *functions*.

Physical constraints. The *physical constraints* define constraints between the *system quantities*. Examples of the *physical constraints* are balances, diffusion laws and (physical) definitions of new *system quantities*. The *physical constraints* are associated to the *system quantities* involved in the *constraint*. These links form the foundation of a model representation called And-Or-Graph in [1]. The *physical do-*

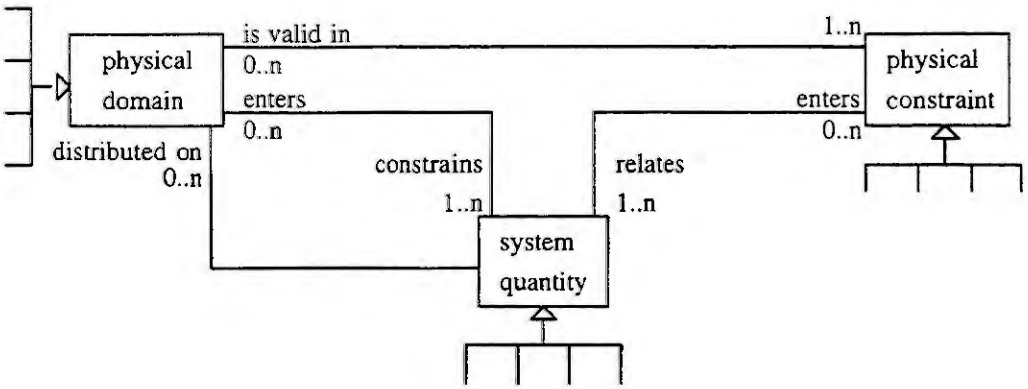


Figure 2: Top level of the partial data model for the physical interpretation concepts.

mains associated with the *constraints* define the scope and application area of the *constraints*. Obviously the *physical constraints* will be linked to *relations* in the mathematical model. The *domain–constraint* association will be mapped to a *support* association between *relations*.

Physical domains. The *physical domains* sum up concepts like volumes and areas which are linked to balance equations. The time interval of a simulation experiment is a *domain* usually defined only implicitly in simulation applications. Sophisticated models will include intervals of substantial coordinates or the Reynold's number dependend validity of a heat transfer law as *domains*. In the mathematical model *domains* are the *support relations* or the integration domain of an integral operator.

4 Summary

A data model was sketched which captures the mathematical as well as physical aspects of models of physical systems. However due to the limited length of this paper certain aspects of the data model had to be excluded. They include the definition of sets with scalar and vector elements which will be associated to the *interface* classes *values* and *functions*.

The *mathematical* partial data model already describes a wide range of mathematical concepts. This will allow reuse of this data model to capture models of other domains, e.g. economics, if an appropriate partial data model is added to capture the new semantics.

References

[1] Bogusch, R. and Marquardt, W., A formal representation of process model equations. Computers and Chemical Engineering, **21** (10), 1997, 1105-1115.

[2] Bronstein, I.N. and Semendjajew, K.A., Taschenbuch der Mathematik. B.G. Teubner Verlagsgesellschaft, Stuttgart, Leipzig, 1991.

[3] Bunge, M. A., Foundations of Physics. Springer-Verlag Berlin Heidelberg New York, 1976.

[4] Gerstlauer, A., Hierlemann, M. and Marquardt, W., On the representation of balance equations in a knowledge based process modeling tool. Paper presented at CHISA'93, Prague, Czech Republic, 1993. (Available at *http://www.lfpt.rwth-aachen.de*)

[5] Marquardt, W., Towards a process modeling methodolgy. In: R. Berber, Ed., Methods of model-based control, NATO-ASI Ser. E, Applied Sciences, Vol. 293, Kluwer Academic Publishers, Dordrecht, 1995.

[6] Marquardt, W., von Wedel, L. and Bayer, B., Perspectives on lifecycle process modeling. Paper presented at FOCAPD'99, Breckenridge, Colorado, U.S.A., 1999. (Available at *http://www.lfpt.rwth-aachen.de*)

INDEX PROBLEMS IN MODELING AND SIMULATION OF FLEXIBLE MECHANICAL SYSTEMS

C. Maffezzoni and P. Rocco
Politecnico di Milano
Dipartimento di Elettronica e Informazione
Piazza Leonardo da Vinci, 32, 20133 Milano - Italy

Abstract. This paper discusses the definition of the index for systems of partial differential and algebraic equations and its role in their numerical solution. Reference is made to the case of a flexible mechanical system (an inextensible cable), whose model is formulated in different, yet dinamically equivalent, ways, with different properties with respect to the feasibility of an accurate numerical integration.

Introduction

A key property of DAE (Differential Algebraic Equations) systems is the index [1]: a DAE whose index is higher than one cannot be reliably integrated by any general purpose numerical solver, since the accuracy of the numerical integration is lost as the stepsize is reduced. This property is just the numerical consequence of a modeling problem, namely that higher index DAEs are ill-posed dynamic models of systems, where explicit or hidden algebraic constraints on state variables are present.

When the unknowns of the model depend on one or more spatial coordinates, as well as on time, PDAE (Partial Differential Algebraic Equations) systems are used to represent the dynamics. The PDAE can then be either numerically integrated simultaneously in space and time, or reduced to a DAE system in time, by way of a discretization of the spatial domain (Method Of Lines, or MOL). In either case it is convenient to check that the PDAE is well posed before going through the numerical integration. Suitable indices should then be defined and used.

In this paper the notion of algebraic index, introduced in [2], is used and extended to the case of nonlinear PDAEs. Reference is then made to the dynamic modeling of inextensible cables, which has been the subject of several papers (see e.g. [3], [4], [5]). In particular in [3] a comprehensive model of an undersea cable is given, where the inextensible case is obtained as the limit case of an elastic cable, setting to zero an elasticity parameter.

Starting from a higher index model of the inextensible cable expressed in cartesian coordinates, a second model, fully equivalent to the one proposed in [3], will be derived, which however is still characterized by a high index (and thus is ill posed). A third model will finally be presented, whose index properties allow integration both in the time and space domains. The methodology used in this paper to reduce the index can be extended to other cases of flexible mechanical systems.

The index of a DAE

Let us first review the definition of the index for a DAE. For this, consider a linear system of equations:

$$A\dot{y}(t)+Cy(t)=f(t)\,, \tag{1}$$

where y is the vector of the n unknowns, A and C are $n\times n$ matrices, with A singular (otherwise the DAE is actually an implicit ODE), and f is a vector of forcing functions. Assume that the DAE is regular, i.e. assume that $\exists\ p_0$ such that the matrix (p_0A+C) is nonsingular. The index of the DAE (1) is defined as the smallest integer n such that all the entries of the following matrix are strictly proper (i.e. they vanish as p approaches infinity):

$$p^{-n}(pA+C)^{-1}\,.$$

Extensions to nonlinear DAEs are feasible through definition of suitable local indices [6].

What makes the index a key property of a DAE system is that it qualifies the accuracy of the numerical integration by reduction of the stepsize. All general purpose DAE codes actually start off the integration with the implicit Euler formula [1], which, ones applied to (1), yields:

$$A(y_k-y_{k-1})/h+Cy_k=f_k\ ,$$

or

$$(pA+C)y_k=pAy_{k-1}+f_k\ ,$$

where $p=1/h$ (h being the stepsize). It is apparent from the above definition that if the index of the DAE is higher or equal to 2, one or more entries of the inverse of the Jacobian $pA+C$ of the discretized equation approach infinity as the stepsize h approaches zero. This is basically the reason why no common DAE solver can reliably integrate higher (>1) index DAEs.

The index of a PDAE

The index of a PDAE has been the subject of a recent paper [2], where it is pointed out that different indices, whose values do not necessarily coincide, may be defined. In this paper we will refer to the concept of algebraic index. Let us therefore consider a linear system of PDAEs, where a one-dimensional spatial domain is assumed:

$$A\dot{y}(s,t)+By'(s,t)+Cy(s,t)=f(s,t)\,, \tag{2}$$

where y is the vector of the n unknowns, $\dot{y}$ and y' are its derivatives with respect to time t and space s, respectively, A, B and C are $n\times n$ matrices, with A or B singular, and f is a vector of forcing functions. Moreover, assume that the PDAE is regular, i.e. assume that $\exists$ (p_0, q_0) such that the matrix (p_0A+q_0B+C) is nonsingular.

The algebraic t-index of the PDAE (2) is the smallest integer n such that all the entries of the following matrix are strictly proper as functions of p:

$$p^{-n}(pA+qB+C)^{-1}\,, \tag{3}$$

while the algebraic s-index is the smallest integer n such that all the entries of the following matrix are strictly proper as functions of q:

$$q^{-n}(pA+qB+C)^{-1}\,. \tag{4}$$

If a nonlinear PDAE is given:

$$F(y,\dot{y},y',f)=0\,, \tag{5}$$

local algebraic indices are computed with reference to the linearized equation (2), where:

$$A=F_{\dot{y}}=\frac{\partial F}{\partial \dot{y}}\;,\quad B=F_{y'}=\frac{\partial F}{\partial y'}\;,\quad C=F_{y}=\frac{\partial F}{\partial y}\,, \tag{6}$$

and the three matrices depend on the linearization point. It is then convenient to define structural algebraic indices for the nonlinear system (5): the structural algebraic (t- or s-) index of (5) is the minimum algebraic index taken by (2), when any non-zero entry of matrices A, B and C computed by (6) in any point is given any finite real value. Obviously the local algebraic index cannot be smaller than the structural index in any point.

The t- and s- indices qualify the accuracy of the numerical integration by reduction of the stepsize in t and in s, respectively. If a MOL integration method is used, thereby discretizing the spatial domain, leaving a DAE in t, the t-index of the PDAE will generally coincide with the index of the DAE, thus qualifying the feasibility of its numerical integration.

The algebraic index does not depend on the boundary conditions of the PDAE. More involved definitions of the index, accounting also for boundary conditions, are discussed in [2].

Models of flexible cables

The concepts discussed above are now applied to the case of inextensible cables. Three different models are given, and for each one the algebraic indices are computed.

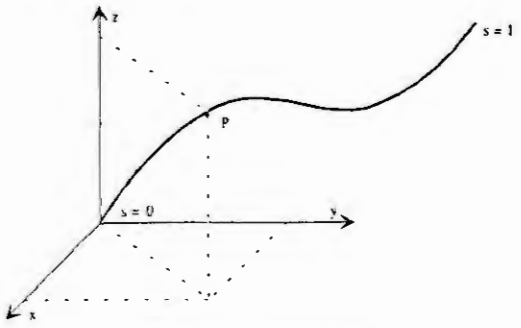


Fig. 1 Cartesian coordinates

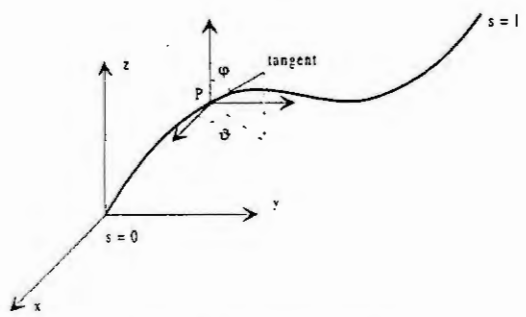


Fig. 2 Angular coordinates

Formulation in cartesian coordinates

Consider a flexible cable in a cartesian space (x, y, z) (Fig. 1) and let s denote the unstretched cable length coordinate: $s=0$ corresponds to the first end of the cable, $s=l$ to the second end. Assume that the point at $s=0$ (first end) is in the origin of the cartesian space. The most natural way to describe the motion of the cable is by way of the functions $x(s, t)$, $y(s, t)$, $z(s, t)$, cartesian coordinates of point P (as in Fig. 1), where t is the time variable.

No elasticity is considered in the cable. The inextensibility of the cable implies the following constraint:

$$(\partial x/\partial s)^2 + (\partial y/\partial s)^2 + (\partial z/\partial s)^2 = 1 \ . \tag{7}$$

The physical modeling of the cable dynamics can be carried out either by formulation of the standard motion equations of an infinitesimal element or by variational arguments, deriving the so called Euler-Lagrange partial derivative equations [7]. In either case, the result can be written in the following vector form:

$$-T'\boldsymbol{w} - T\boldsymbol{w}' + \rho\dot{\boldsymbol{v}} = \boldsymbol{f} \qquad \dot{\boldsymbol{r}} - \boldsymbol{v} = 0 \qquad \boldsymbol{r}' - \boldsymbol{w} = 0 \qquad \boldsymbol{w}^T\boldsymbol{w} = 1 \ , \tag{8}$$

where the three dimensional vectors $\boldsymbol{r} = [x, y, z]$, $\boldsymbol{v}$, $\boldsymbol{w}$ and $\boldsymbol{f}$ have been introduced. Vector $\boldsymbol{f} = \boldsymbol{f}(s, t)$ is made up by the external forces per unit of length acting at time t on the cable, in the point associated to the coordinate s; ρ is the mass per unit length of the cable; the scalar variable T is the tension of the cable.

Defining with $\boldsymbol{I}_t$ and $\boldsymbol{I}_s$ the matrices whose entries contain the smallest integers such that the corresponding entry of matrices (3) or (4), respectively, are strictly proper, we obtain (assigning random numbers to the nonzero entries of $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$, and arranging the variables in the vector $\boldsymbol{y} = [\boldsymbol{r}, \boldsymbol{v}, \boldsymbol{w}, T]$):

$$\boldsymbol{I}_t = \begin{bmatrix} 0&0&0&0&0&0&1&1&1&1 \\ 0&0&0&0&0&0&1&1&1&1 \\ 0&0&0&0&0&0&1&1&1&1 \\ 0&0&0&1&1&1&2&2&2&2 \\ 0&0&0&1&1&1&2&2&2&2 \\ 0&0&0&1&1&1&2&2&2&2 \\ 0&0&0&0&0&0&1&1&1&1 \\ 0&0&0&0&0&0&1&1&1&1 \\ 0&0&0&0&0&0&1&1&1&1 \\ 1&1&1&2&2&2&3&3&3&3 \end{bmatrix}, \quad \boldsymbol{I}_s = \begin{bmatrix} 0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0&0&0 \\ 0&0&0&0&1&0&0&0&0&0 \\ 0&0&0&0&0&1&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&1 \\ 0&0&0&0&0&0&0&0&0&1 \\ 0&0&0&0&0&0&0&0&0&1 \\ 0&0&0&0&0&0&0&0&0&1 \end{bmatrix} .$$

The structural t-index is therefore 3, the structural s-index is 1.

First formulation in angular coordinates

An alternative way to describe the motion of the system is to give the orientation of the vector tangent to the cable at any point P. Consider a set of spherical coordinates, expressed with reference to a cartesian frame, parallel to (x, y, z), but with origin in point P (Fig. 2). The orientation of the tangent vector can be expressed by the two angles ϑ and φ, and the motion of the cable is completely described once the functions $\vartheta(s, t)$ and $\varphi(s, t)$ are given.

On the other hand, the cartesian coordinates (x, y, z) of point P are expressed in terms of the new coordinates ϑ and φ by way of the following differential equations, as it can be easily verified:

$$\partial x/\partial s = \cos\vartheta \sin\varphi \ , \quad \partial y/\partial s = \sin\vartheta \sin\varphi \ , \quad \partial z/\partial s = \cos\varphi \ .$$

As a consequence, the constraint (7) is identically satisfied in ϑ and φ for every value of the couple (s, t).

The equations of motion can be derived from the equations of the cartesian model. First introduce the following vectors and matrices (where $\eta = [\vartheta, \varphi]$):

$$\boldsymbol{E}(\eta) = \begin{vmatrix} \cos\vartheta \sin\varphi \\ \sin\vartheta \sin\varphi \\ \cos\varphi \end{vmatrix}, \quad \Gamma(\eta) = \begin{vmatrix} -\sin\vartheta \sin\varphi & \cos\vartheta \cos\varphi \\ \cos\vartheta \sin\varphi & \sin\vartheta \cos\varphi \\ 0 & -\sin\varphi \end{vmatrix} .$$

Then replace $\boldsymbol{w}$ with $\boldsymbol{E}(\eta)$ in the first equation of system (8). This yields:

$$-T'\boldsymbol{E}(\eta) - T\Gamma(\eta)\eta' + \rho\dot{\boldsymbol{v}} = \boldsymbol{f} \qquad \dot{\boldsymbol{r}} - \boldsymbol{v} = 0 \qquad \boldsymbol{r}' - \boldsymbol{E}(\eta) = 0 \ .$$

At this point the derivative $\dot{\boldsymbol{r}}$ is replaced by a fictitious algebraic vector variable $\boldsymbol{u}$, while the third vector equation is differentiated with respect to time. In doing so, the derivative of vector η with respect to time appears. The new system of equations becomes:

$$-T'\boldsymbol{E}(\eta) - T\Gamma(\eta)\eta' + \rho\dot{\boldsymbol{v}} = \boldsymbol{f} \qquad \boldsymbol{u} - \boldsymbol{v} = 0 \qquad \boldsymbol{r}' - \boldsymbol{E}(\eta) = 0 \qquad \boldsymbol{u}' - \Gamma(\eta)\dot{\eta} = 0$$

The second vector equation is a trivial algebraic identity and can be eliminated, the third one is an explicit differential equation in s and can be ignored, since $\boldsymbol{r}$ does not appear in the remaining equations.

The final expression of the system of equations is therefore:

$$-T'\boldsymbol{E}(\eta) - T\Gamma(\eta)\eta' + \rho\dot{\boldsymbol{v}} = \boldsymbol{f} \qquad \boldsymbol{v}' - \Gamma(\eta)\dot{\eta} = 0 \ , \tag{9}$$

and is fully equivalent to the inextensible case in [3]. Arranging the variables in the vector $y = [v, \eta, T]$, the matrices I_t and I_s defined as above turn out to be now:

$$I_t = \begin{bmatrix} 0&0&0&1&1&1 \\ 0&0&0&1&1&1 \\ 0&0&0&1&1&1 \\ 0&0&0&0&0&0 \\ 0&0&0&0&0&0 \\ 1&1&1&2&2&2 \end{bmatrix}, \quad I_s = \begin{bmatrix} 0&0&0&0&0&0 \\ 0&0&0&0&0&0 \\ 0&0&0&0&0&0 \\ 0&0&0&0&0&0 \\ 0&0&0&0&0&0 \\ 0&0&0&0&0&0 \end{bmatrix},$$

which means that the structural t-index is 2 and the structural s-index is 0.

Second formulation in angular coordinates

The procedure that yielded system (9) is now iterated. The derivative $\dot{v}$ is therefore replaced by the fictitious algebraic vector variable $\boldsymbol{a}$, while the second vector equation is differentiated with respect to time. In doing so, the second derivative of vector η with respect to time appears. In order to remain consistent with the general expression (5) of the PDAE, a new vector variable $\omega = [\omega_\vartheta, \omega_\varphi]$ is introduced, equal to the first derivative $\dot{\eta}$. The new system of equations thus becomes:

$$-T'\boldsymbol{E}(\eta) - T\boldsymbol{\Gamma}(\eta)\eta' + \rho\boldsymbol{a} = f \qquad \boldsymbol{a}' - \boldsymbol{\Gamma}(\eta)\dot{\omega} - \left(\omega_\vartheta \boldsymbol{R}_1(\eta) + \omega_\varphi \boldsymbol{R}_2(\eta)\right)\omega = 0 \qquad \dot{\eta} - \omega = 0 \ ,$$

where:

$$\boldsymbol{R}_1(\eta) = \begin{vmatrix} -\cos\vartheta\sin\varphi & -\sin\vartheta\cos\varphi \\ -\sin\vartheta\sin\varphi & \cos\vartheta\cos\varphi \\ 0 & 0 \end{vmatrix}, \ \boldsymbol{R}_2(\eta) = \begin{vmatrix} -\sin\vartheta\cos\varphi & -\cos\vartheta\sin\varphi \\ \cos\vartheta\cos\varphi & -\sin\vartheta\sin\varphi \\ 0 & -\cos\varphi \end{vmatrix},$$

and the explicit differential equation in s:

$$v' - \boldsymbol{\Gamma}(\eta)\omega = 0 \ ,$$

has been ignored, since v does not appear in the remaining equations. Defining the vector $y = [\boldsymbol{a}, \eta, \omega, T]$, the matrices I_t and I_s result:

$$I_t = \begin{bmatrix} 1&1&1&1&1&1&0&0 \\ 1&1&1&1&1&1&0&0 \\ 1&1&1&1&1&1&0&0 \\ 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \\ 1&1&1&1&1&1&0&0 \end{bmatrix}, \ I_s = \begin{bmatrix} 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&0&1 \\ 0&0&0&0&0&0&0&0 \end{bmatrix},$$

which means that now both the structural t-index and the structural s-index are 1.

Conclusions

Three models of the same infinite dimensional system, an inextensible cable, have been discussed. Since only the third one has structural t-index less than two, it is the only one that can be safely reduced to a DAE system, for simulation with a general purpose numerical code. Experiences with a DAE solver (gPROMS [8]), endowed with an embedded tool for discretization of a spatial domain, confirmed that the first two models cannot be integrated, particularly when sudden changes in the external inputs (e.g. a force on the free end), implying sudden reductions of the stepsize, are applied.

References

[1] K.E. Brenan, S.L. Campbell and L.R. Petzold (1989), Numerical solution of initial-value problems in differential algebraic equations, North-Holland.

[2] S.L. Campbell and W. Marszalek (1999), "The index of an infinite dimensional implicit system", Math. Comp. Modell. of Dyn. Sys., **5**, pp. 18-42.

[3] C.M. Ablow and S. Schechter (1983), "Numerical simulation of undersea cable dynamics", Ocean Eng., **10**, pp. 443-457.

[4] S. Huang (1994), "Dynamic analysis of three-dimensional marine cables", Ocean Eng., **21**, pp. 587-605.

[5] Y. Sun, J.W. Leonard, R.B. Chiou (1994), "Simulation of unsteady oceanic cable deployment by indirect integration with suppression", Ocean Eng., **21**, pp.243-256.

[6] S.L. Campbell and C.W. Gear (1995), "The index of general nonlinear DAEs", Num. Math., **72**, pp.173-196.

[7] H. Goldstein (1980), Classical mechanics, Addison Wesley.

[8] M. Oh and C.C. Pantelides (1996), "A modelling and simulation language for combined lumped and distributed parameter systems", Comp. Chem. Eng., **20**, pp. 611-633.

NUMERICAL SIMULATION OF TUBULAR REACTORS : SOME PROPERTIES OF THE ORTHOGONAL COLLOCATION

L. Lefèvre[+], D. Dochain[*] and A. Magnus[△]

[+]ESISAR, GSys unit, 50 rue Barthelemy de Laffemas, BP54, 26902 Valence, France
[*]CESAME, UCL, av. G. Lemaître 4, 1348 Louvain-La-Neuve, Belgium
[△]Numerical Analysis & Applied Mathematics Unit, UCL,Louvain-la-Neuve, Belgium

Abstract. In this paper, we analyse some properties of the orthogonal collocation in the context of its use for reducing PDE (Partial Differential Equations) chemical reactor models for numerical simulation and/or control design. The approximation of the first order derivatives are analyzed with respect to the transfer of the stability properties of the hydrodynamics from the PDE model to its approximated ODE (ordinary differential equations) model. Then the choice of the collocation points as zero of Jacobi polynomial is analyzed and interpreted as an optimal choice with respect to a weighted norm.

1 Introduction

The dynamics of tubular reactors are described by partial differential equations (PDE's) derived from mass and energy balances (e.g. [4]). Either for numerical simulation or control design, the PDE's model is commonly reduced to ordinary differential equations (ODE's) by using approximation methods (e.g. finite differences, orthogonal collocation)[5]. It may be difficult generally speaking to know the connection between the original distributed parameter model and its (finite dimensional) discretised version, and as mentioned in [8], the dynamical properties of both models may be different. This is largely due to the lack of knowledge about the properties of such approximation method. In this paper we analyze some properties of the orthogonal collocation approximation method. After the introduction of the general dynamical PDE model for tubular reactors, Section 3 briefly presents the orthogonal collocation as a method for reducing PDE models into ODE ones and analyzes the structural stability properties of the matrix characterizing the reduction of the first order space derivative. Section 4 considers the choice of collocation points as an error minimization problem and shows that the choice of zeros of Jacobi polynomials corresponds to an optimal choice for a weighted norm. The detailed results can be found in ([6]).

2 Dynamical Models of Chemical Tubular Reactors

Let us consider a tubular reactor in which N_R non-isothermal reactions take place involving N_C components (reactants and products), and with axial dispersion (diffusion). Then the dynamical model is readily derived by using mass and energy balances and can be written in the following matrix form :

$$\frac{\partial x}{\partial t} = -u\frac{\partial x}{\partial z} + D_a\frac{\partial^2 x}{\partial z^2} + \tilde{K}r(x) + U \tag{1}$$

$$\text{with : } x = \begin{pmatrix} C \\ T \end{pmatrix},\ \tilde{K} = \begin{pmatrix} K \\ -\frac{\Delta H^T}{\rho C_p} \end{pmatrix}, U = \begin{pmatrix} -\frac{4h}{\rho d_\rho C_p}(T_w - T) \\ 0 \end{pmatrix},\ D_a = \begin{pmatrix} D_{ma} I_{N_f} & 0 \\ 0 & \frac{\lambda_{ea}}{\rho C_p} \end{pmatrix} \tag{2}$$

and where T is the temperature (K), C is the component concentration vector ($\frac{mol}{l}$), t (s) and z (m) the time and space variables, λ_{ea} and D_{ma} the axial energy and mass dispersion coefficients ($\frac{kJ}{msK}$), u the superficial fluid velocity ($\frac{m}{s}$), L the reactor length (m), ΔH the reaction heat vector ($\frac{kJ}{mol}$), ρ the fluid density ($\frac{kg}{m^3}$), C_p the specific heat ($\frac{kJ}{kgK}$), $r(x)$ the reaction rate vector ($\frac{mol}{ls}$), K the yield coefficient matrix and h, d and T_w the wall heat transfer coefficient ($\frac{kJ}{m^2Ks}$), the reactor diameter (m) and coolant temperature (K) respectively (see e.g., [3] [4]). An important particular situation is the plug flow reactor, i.e. when $D_a = 0$: the model is then hyperbolic (it is parabolic otherwise). The model is completed with the following two boundary conditions (with x_{in} the influent value of x) [2] :

$$D_a \left.\frac{\partial x}{\partial z}\right|_{z=0} = -u(x_{in} - t) - x(z = 0, t)); \qquad \left.\frac{\partial x}{\partial z}\right|_{z=L} = 0 \tag{3}$$

3 Reduction of the partial differential equations

The principle of the orthogonal collocation method is to search an approximation in the form of a finite series $x^*(z,t) = \sum_{i=0}^{N} c_i(t) \cdot l_i^{(N)}(z)$, where x^* denotes the approximation, N the reduction order, $c_i(t)$ time-varying coefficients, and $l_i^{(N)}(z)$ N^{th} order Lagrange interpolation polynomials :

$$l_i^{(N)}(z) = \prod_{\substack{j=0 \\ \neq i}}^{N} \frac{z - z_j}{z_j - z_i} \tag{4}$$

where $z_0, \ldots, z_N \in [0, L]$, the interpolation (or "collocation") points are parameters of the method, as well as N, the order of the reduction. The unknown time-varying coefficients $c_i(t)$ are chosen such that the approximated solution is the exact one at the collocation points. Since we have $l_i(z_j) = \delta_{ij}$, this means : $c_i(t) = x^*(z_i, t) = x(z,t)|_{z=z_i} \quad \forall i \in \{0, \ldots, N\}$. For the system (1), we obtain the following set of ordinary differential equations :

$$\frac{dx_d}{dt} = (-uC_1 + D_a C_2)x_d + \tilde{K}_f r(x_d) + U_d + (-uc_1 + D_a c_2)x^*(z_0, t) \tag{5}$$

where x_d denotes the vector of the component concentrations and temperature at the collocation points, that is :

$$x_d^T = \left(x_1^*(z_1), \ldots, x_1^*(z_r); \cdots; x_{N_C}^*(z_1), \ldots, x_{N_C}^*(z_r); T^*(z_1), \ldots T^*(z_r)\right) \tag{6}$$

U_d and $r(x_d)$ are the external heat exchange rate and reaction rate vectors computed at the collocation points, and listed in the same order than x_d. C_1 and C_2 are $(N_R + 1) \times (N_R + 1)$ block diagonal matrices where every block of the diagonal are identical, and are the discretised matrices operator obtained from the reduction of, respectively, the convection operator ($\frac{\partial}{\partial z}$), and the dispersion operator ($\frac{\partial^2}{\partial z^2}$), that is :

$$C_k = \begin{pmatrix} \tilde{C}_k & 0 & \cdots & 0 \\ 0 & \tilde{C}_k & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \tilde{C}_k \end{pmatrix} \; ; \; c_k = \begin{pmatrix} \tilde{c}_k \\ \tilde{c}_k \\ \vdots \\ \tilde{c}_k \end{pmatrix} \; ; \; k = 1, 2 \tag{7}$$

$$\tilde{C}_k = [\frac{d^k l_i^{(N)}(z_j)}{dz}]_{\{i,j=1,\ldots,r\}} \; ; \; \tilde{c}_k = [\frac{d^k l_0^{(N)}(z_j)}{dz}]_{\{j=1,\ldots,r\}} \tag{8}$$

The value of the variables x at the boundaries can be computed from the boundary conditions (3) by introducing the orthogonal collocation expansion for x^* introduced hereabove.

The term $(-uC_1 + D_a C_2)x_d$ approximates the hydrodynamics term of the PDE model, $-u\frac{\partial x}{\partial z} + D_a \frac{\partial^2 x}{\partial z^2}$. It is obvious that due to flow direction in the reactor, the equation $\frac{\partial x}{\partial t} = -u\frac{\partial x}{\partial z} + D_a \frac{\partial^2 x}{\partial z^2}$ is asymptotically stable, or in other hands, the hydrodynamics are stable. This property is easy to check by using standard PDE methods. Note that for the approximated models because of the structure of C_1 and C_2, it is enough to look at the matrices $\tilde{C}_1$ and $\tilde{C}_2$. If we consider finite differences, the eigenvalues of the (bi- and tri- diagonal) matrices $\tilde{C}_1$ and $\tilde{C}_2$ are easy to compute : it is then routine to check that the term $(-uC_1 + D_a C_2)x_d$ in finite difference is stable (see e.g. [3]). For the orthogonal collocation, the matrices $\tilde{C}_1$ and $\tilde{C}_2$ are full, and this renders the verification more intricate. For simplicity we shall only concentrate on the matrix C_1 (plug flow reactor), i.e. on the term $-uC_1 x_d$. The analysis is rather involved and is based on the properties of the Lagrange polynomials used as the base functions. It can be shown that the matrix $\tilde{C}_r$ can be written as follows (see [1]):

$$\tilde{C}_r = B_{22} Z_{22}^r B_{22}^T \tag{9}$$

$$\text{with :} \quad B = \begin{pmatrix} b_0 & 0 \\ 0 & B_{22} \end{pmatrix}, Z^r = \begin{pmatrix} Z_{11}^r & Z_{12}^r \\ Z_{21} & Z_{22}^r \end{pmatrix}, \quad B = diag(b_j), \; b_j = \prod_{k=0, \neq j}^{N} (z_j - z_k) \tag{10}$$

$$Z = (z_{jk}), \quad z_{jk} = \sum_{i=0, \neq j}^{N} \frac{1}{z_j - z_i} \quad if \; j = k \quad \text{or} \; \frac{1}{z_j - z_k} \quad if \; j \neq k \tag{11}$$

B_{22} can be viewed as a similarity transformation : therefore the eigenvalues for $\tilde{C}_\tau$ and Z_{22}^τ are similar. For the plug flow reactor case, it can be shown that the real parts of the eigenvalues of Z_{22} are all positive up to $N = 4$. However for larger values of N ($N > 4$), the conjecture that the real parts of the eigenvalues of Z_{22} are positive is wrong. An important guideline at this point consists of checking the eigenvalues of $(-u\tilde{C}_1 + D_a\tilde{C}_2)$ before implementing the orthogonal collocation approximation.

4 Theoretical accuracy of the collocation method

The application of orthogonal collocation to the approximation of distributed parameter models. Its implementation requires to choose [7] the number of collocation points and the location of the collocation points. A classical choice of collocation points in chemical engineering is to take them as zeros of orthogonal polynomials, usually of N^{th} order monic Jacobi polynomials $p_N^{(\alpha,\beta)}$. Choosing zeros of classical orthogonal polynomials as collocation points makes orthogonal collocation approximations able to integrate exactly polynomials up to order $2N-1$ by means of quadrature formulas [10], which actually is the maximum order of accuracy reachable with such N^{th} order approximations. In this sense, this choice can be considered as optimal, and, in practice, it provides results comparable with those obtained from Galerkin's method [10] (often considered as a reference but it requiring a huge computational effort).

Once this generic choice of collocation points agreed, we may wonder more specifically why to choose zeros of Jacobi polynomials. One possible reason to do so is that this choice leads to a numerical method which can be "tuned" with two parameters : α and β. For instance (see [5]), α small (resp. > 1) and $\beta > 1$ (resp. small) tend to concentrate the collocation points close to the reactor output (resp. input). This property provides an intuitive tuning for the method and has been extensively used by process engineers to define qualitative and experimental-based rules governing the appropriate use of the method in each specific application. However, we will not, in this paper, restrict our a priori choice to Jacobi polynomials, but will consider general classical orthogonal polynomials.

Let us define the interpolation error by : $e_N(z,t) := x(z,t) - \sum_{i=0}^{N} x_i(t) \cdot p_i(z)$. This equation suggests that we can handle the effect of large variations (typically, due to the presence of "hot spots" i.e. large and concentrated variations in the spatial profiles of the temperature and/or concentrations) on the interpolation error by choosing suitable collocation points. Indeed, we may wish to choose the $N-1$ interior collocation points solutions of the following problem :

$$\min_{z_i \in [-1,+1] \ \forall i \in 1,\ldots,N} \left(\left\| \prod_{j=0}^{N} |(z - z_j)\, w_\infty(z)| \right\|_\infty \right). \tag{12}$$

where the weight $w_\infty(z)$ is supposed to be large around the hot spots, and small everywhere else. Doing this, we keep the interpolation error small where it would have been large if we had considered an "uniform" choice of collocation points (zeros of a Chebyshev polynomial). Actually this choice follows the classical intuitive choices of collocation points, since it increases collocation points around hot spots. However, a compromise has to be found since the number of collocation points is limited (it is also the size of the differential system to solve) and since a too high concentration of collocation points around hot spots leads to poor approximation anywhere else. This compromise is automatically handled if collocation points are chosen according to (12).

Let us remark that the problem of finding the collocation points $z_1, \ldots, z_{N-1}$ which minimizes the product $\prod_{j=0}^{N} (z - z_j)\, w_\infty(z)$ is equivalent the problem of finding the monic polynomial $\prod_{j=0}^{N} (z - z_j)$ of minimal weighted uniform norm, defined in the following way:

$$\forall f \in C([a,b]), \ \|f\|_{w_\infty} := \sup_{x \in [a,b]} |f(z) \cdot w_\infty(z)| \tag{13}$$

Then note that this last problem is equivalent to the problem of finding the best N^{th} order (not monic) approximation polynomial for the function $-z^{N+1}$, in the sense of the weighted norm $\|\cdot\|_{w_\infty}$, that is finding the polynomial $\tilde{p}_{N-1}$ such that

$$\left\| (-z^{N+1}) - \tilde{p}_{N-1} \right\|_{w_\infty} = \left\| - \prod_{j=0}^{N} (z - z_j) \right\|_{w_\infty} \tag{14}$$

is minimal. The solution of this last approximation problem may be characterized by using a generalization of the Chebyshev equi-oscillation theorem presented hereafter.

Theorem 1 (Alternation theorem) *If* $f \in C([a,b])$; $\hat{p} \in \mathcal{P}_N (N > 0)$ *where* $\mathcal{P}_N$ *denotes the space of* $N^{th.}$ *order polynomial with real coefficients defined on* $[-1,+1]$, **Then** $\hat{p}$ *is the best approximation of* f, *in* $\mathcal{P}_N$, *in the sense of the norm* $\|\cdot\|_{w_\infty}$ **iff.** *There is a set of at least* $(N+2)$ *points such that the function* $\hat{e} := (f - \hat{p}) \cdot w_\infty$ *equi-oscillates and reaches its extrema values at each of these points, that is :*
iff. $\exists (z_i)_{i \in 0,\ldots,N+1}$ *such that* $a \leq z_0 < z_1 < \ldots < z_{N+1} \leq b$ *and* $\hat{e}(z_i) = \sigma(-1)^i \|\hat{e}\|_\infty$ *with* $\sigma = 1$ *or* -1

In the case of general weights, only asymptotic results (that is, results when the polynomial order N tends to infinity) are available on equi-oscillation properties. From this theory [9], we point out the following result : *Assume* $w(z) := \frac{d\alpha}{dz}$ *is the weight used to define a family of orthonormal polynomial with respect to the inner product* $(\cdot|\cdot)_{L^2_{d\alpha}}$. *Then the quantity* $\sqrt{\sqrt{1-z^2}w(z)}p_N(z)$ *(where* p_N *is the* $N^{th.}$ order *polynomial orthonormal polynomial associated to the weight* w*) tends to asymptotically equi-oscillate on the interval* $[-1,+1]$, *when* $N \to \infty$.

If we assume that N is large enough, we can then conclude that $\sqrt{\sqrt{1-z^2}w(z)}p_N(z)$ nearly equi-oscillates. Indeed, since the operator of the best polynomial approximation is continuous, we may conclude that this $p_N(z)$ (which nearly equi-oscillates) is a good approximation of the optimal polynomial $\hat{p}_N(z)$, which minimizes the norm $\|p\|_{w_\infty}$ with : $w_\infty(z) := \sqrt{\sqrt{1-z^2}w(z)}$.

At this point, it may be interesting to investigate the meaning of classical choices of collocation points in order to point out what implicit norm minimization is made when using it. For instance, in the case where the collocation points are zeros of Jacobi polynomials, we know that : $w(z) := (1-z)^\alpha(1+z)^\beta$. Hence we know that using these collocation points we will get an approximated solution which (theoretically) minimizes an interpolation error weighted by the function:

$$w_\infty(z) = \sqrt{\sqrt{1-z^2}(1-z)^\alpha(1+z)^\beta} \text{ (with } \alpha,\beta \geq \frac{-1}{2}) = (1-z)^{\frac{2\alpha+1}{4}} \cdot (1+z)^{\frac{2\beta+1}{4}} \qquad (15)$$

Many shapes may be reached with weights of the form (15). They allow to emphasize the interpolation error from one side to the other of the reactor, according to the values of the tuning parameters α and β.

References

[1] Calogero F., *Some applications of a convenient finite-dimensional matrix representation of the differential operator*, Rendiconti del Seminario Matematico, 23-61 (1985).

[2] Danckwerts P.V., *Continuous flow systems. Distribution of residence times*, Chem. Eng. Sci., 2 (1), 1-13 (1953).

[3] Dochain D., *Contribution to the Analysis and Control of Distributed Parameter Systems with Application to (Bio)chemical Processes and Robotics*, Thèse d'Agg. Ens. Sup., UCL, Belgium (1994).

[4] Feyo de Azevedo S., M.A. Romero-Ogawa and A.P. Wardle, *Modelling of Tubular Fixed-Bed Catalytic Reactors : a Brief Review*, Trans. I.Chem.E., 68(A), 2-8 (1990).

[5] Georgakis C., R. Aris and R. Amundson, *Studies in the control of tubular reactors - I. General considerations*, Chem. Eng. Sci., 32, 1359-1369 (1977).

[6] Lefèvre L., D. Dochain, S. Feyo de Azevedo and A. Magnus, *Analysis of the orthogoanl collocation method applied to the numerical integration of chemical reactor models*, subm. for publ. (1999).

[7] Michelsen M.L. and J.Villadsen, *A Convenient Computational Procedure for Collocation Constants*, The Chemical Engineering Journal, 4, 64-68 (1972).

[8] Ray W.H., *Advanced Process Control*, Butterworths, Boston (1981).

[9] Walter Van Assche, *Asymptotics for Orthogonal Polynomials*, Springer Verlag (1987)

[10] Villadsen J.and M.L. Michelsen, *Solution of Differential Equation Models by Polynomial Approximation*, Prentice-Hall, Englewood Cliffs (1978).

SET-MEMBERSHIP EQUALIZATION

Cécile Durieu*, Eric Walter◊, Sylvie Marcos◊, Odile Macchi◊
* LESiR CNRS, ENS Cachan, 94235 Cachan, France
◊ L2S, CNRS-Supélec-Université de Paris-Sud, 91192 Gif-sur-Yvette, France

Abstract: A new approach is presented to the equalization of telecommunication channels. It is based on set-membership state estimation techniques. These techniques make it easy to take into account the fact that the symbols emitted belong to a finite alphabet.

Keywords: bounded noise, equalization, ellipsoidal bounding, set estimation, state estimation, telecommunications.

Introduction

Equalization is a problem of considerable importance is the context of high-speed digital telecommunications, because intersymbol interference complicates the recovery of the symbols emitted. Under the classical assumption that the channel can be described as a finite impulse response filter with an additive noise, one can write

$$y_t = \mathbf{c}^{\mathrm{T}}\mathbf{x}_t + \gamma_t, \tag{1}$$

where y_t is the sample of the output signal received at time t, $\mathbf{c}$ is the impulse response of the channel, γ_t is the output error resulting from measurement noise and model error, and $\mathbf{x}_t$ consists of the last n symbols emitted

$$\mathbf{x}_t = (d_t, d_{t-1}, \cdots, d_{t-n+1})^{\mathrm{T}}. \tag{2}$$

Each of the symbols d_t belongs to a known and bounded alphabet with m values. Since $\mathbf{x}_t$ is a sliding window, it satisfies the following linear discrete-time equation

$$\mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathbf{v}d_{t+1}, \tag{3}$$

where the $n \times n$ shift matrix $\mathbf{A}$ and $\mathbf{v}$ are known, and given by

$$\mathbf{A} = \begin{bmatrix} 0 & \cdots & \cdots & \cdots & 0 \\ 1 & \ddots & & & \vdots \\ 0 & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 & 0 \end{bmatrix}, \mathbf{v} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}. \tag{4}$$

Equalization is the estimation of the sequence of symbols emitted $\{d_t\}$ from the sequence of symbols received $\{y_t\}$. Assuming that the impulse response $\mathbf{c}$ is known and that the output error γ_t is bounded, it can be viewed as a bounded-error state-estimation problem.

Ellipsoidal outer-bounding techniques for guaranteed set-membership state estimation are first briefly recalled. A new adaptation of these techniques in the context of equalization with a finite alphabet and a bounded output error is then presented. Finally, the resulting methodology is illustrated.

Ellipsoidal bounding

Set-membership estimation is an attractive alternative to conventional approaches such as least squares and Kalman filtering. It can be traced back to the late sixties [1], [12], and has received a lot of attention worldwide, see, *e.g.*, the survey papers [2], [3], special issues of journals [13], [10], book [9] and the references therein. For applications in the context of channel equalization see, for example, [7], [6]. The fundamental hypothesis of set-membership estimation is that the output errors and state perturbations belong to known compact sets, with no other hypotheses on their distributions. This should be contrasted with classical stochastic approaches where the noise is assumed to be random, and usually white and Gaussian. Set-membership estimation aims to characterize the set of all parameter or state vectors that are consistent with the data, model structure and noise bounds. This feasible set can be viewed as a 100% confidence region for the parameter or state vector. All elements of this feasible set are then considered as solutions of the estimation problem.

In general the feasible set is extremely complicated. This is why it is customary to characterize this set by computing a simpler set that encloses it. Among the set-membership estimators, those using outer ellipsoids have received special attention. After enclosing the compact error sets in ellipsoids, they compute the smallest ellipsoid (in a sense to be specified) that can be guaranteed to contain the feasible set. In the context of real-time equalization, only recursive algorithms can be considered. Classically, the quality of the approximation is measured by the volume of the outer ellipsoid, proportional to the square of the product of the lengths of its axes and a natural measure of size. This corresponds to the *determinant criterion*. An alternative measure of size is also considered, namely the sum of the squares of the lengths of the semi-axes, which corresponds to the *trace criterion*. In the context of state estimation (or parameter tracking), at each step, recursive ellipsoidal bounding techniques alternate prediction and correction phases, as in a conventional Kalman filter.

Assume that, at time t, the feasible set is outer-bounded by the ellipsoid $\widehat{\mathcal{E}}_{t/t}$. Let $\mathcal{M}_{t+1}$ be the Minkowski sum of the ellipsoid resulting from the evolution of $\widehat{\mathcal{E}}_{t/t}$ due to the deterministic part of the state equation and of the ellipsoid containing the state noise. At the *prediction step*, $\widehat{\mathcal{E}}_{t+1/t}$ is computed as the smallest ellipsoid containing $\mathcal{M}_{t+1}$. At the *correction step*, an ellipsoid $\widehat{\mathcal{E}}_{t+1/t+1}$ is computed as the smallest ellipsoid containing the intersection of $\widehat{\mathcal{E}}_{t+1/t}$ with the set $\mathcal{S}_{t+1}$ of all states consistent with the measurement y_{t+1}. $\mathcal{S}_{t+1}$ is the strip defined by $\left|y_{t+1} - \mathbf{c}^{\mathrm{T}}\mathbf{x}_{t+1}\right| \leqslant \gamma_{t\,\max}$, where $\gamma_{t\,\max}$ is the known upper bound of $|\gamma_t|$. $\mathcal{S}_{t+1}$ can be viewed as a degenerate unbounded ellipsoid. During the prediction and correction steps, small may be understood in the sense of the determinant or trace criterion. Note that $\mathcal{S}_{t+1}$ is orthogonal to $\mathbf{c}$, which does not depend on t; all strips are therefore parallel to one another. Moreover, the width of the strip is constant if $\gamma_{t\,\max}$ is independent of t. For more details on the computations involved in the prediction and correction steps, see, *e.g.*, [5], [4].

Equalization

Taking advantage of the special structures of $\mathbf{A}$ and $\mathbf{v}$ (4) makes the computation of the prediction step particularly easy. Indeed, $\mathbf{A}$ is a shift matrix and the ellipsoid containing the state noise is a segment, which is a degenerated flat ellipsoid. If any component of $\mathbf{x}_t$ but the last one is known without ambiguity, $\widehat{\mathcal{E}}_{t/t}$ will then be flat; it may even be reduced to a point. As a result its volume will be zero, and the same will hold true for $\widehat{\mathcal{E}}_{t+1/t}$. The determinant criterion then becomes meaningless, unless the dimension of the problem is decreased by restricting the state vector to its ambiguous components. This will be performed before any prediction step. Even in the case of the trace criterion, it can be expected to simplify computation.

Just after the correction step and before returning to the prediction step, the finite nature of the alphabet of symbols will be used to improve the precision of the set estimation. Ideally, one may intersect $\widehat{\mathcal{E}}_{t+1/t+1}$ with the set consisting of the m^n isolated points defined by the finite alphabet. In practice, this would entail testing whether each of these points belongs to $\widehat{\mathcal{E}}_{t+1/t+1}$, an operation of exponential complexity. Moreover, whenever the set resulting from the intersection is not a singleton, one would have to track the evolution of each candidate solution individually, which would have a complexity similar to that of the Viterbi algorithm [11]. To keep computation manageable for large values of m and n, we have chosen instead to use the following decision rule, to be inserted between the correction step and prediction step of the ellipsoidal outer-bounding algorithm.

Decision rule. As long as it reduces the ambiguity and time allows, alternate the following two steps.
Step 1: Project $\widehat{\mathcal{E}}_{t/t}$ onto each of the $(0, d_i)$ axes of the state space associated with the still ambiguous symbols, and test whether only one value belongs to the resulting segment. In this case, the value of the symbol emitted is known without ambiguity.
Step 2: Replace $\widehat{\mathcal{E}}_{t/t}$ by its intersection with the planes associated with all symbols that are known without ambiguity.
Remark 1: If $\widehat{\mathcal{E}}_{t/t}$ has been modified at Step 2, then its volume is now zero and the reduction of the dimension of the state space alluded to during the presentation of the prediction step must be performed.
Remark 2: If time allows, still ambiguous symbols of $\mathbf{x}_{t-1}$ may be processed by the decision rule.

Illustration

Consider the case where only two symbols can be emitted, $d_t \in \{-1, 1\}$, and the output errors γ_t are uniformly distributed in $[-\gamma_{t\,\max}, \gamma_{t\,\max}]$. The signal-to-noise ratio (SNR) is then equal to $3\,\|\mathbf{c}\|^2 / \gamma_{t\,\max}^2$.

Figure 1 presents the projection of $\widehat{\mathcal{E}}_{t/t}$, to which $\mathbf{x}_t$ belongs, onto the plane associated with the last two symbols emitted, namely d_t and d_{t-1}. All possible combinations of values are represented by crosses and the (unknown) truth is indicated by a star. Here, d_{t-1} turns out to be known, so $\widehat{\mathcal{E}}_{t/t}$ is flat and its projection onto the plane of Figure 1 is a segment. Figure 2 presents the result of one prediction step from the situation described by Figure 1, again projected onto the plane of the last two symbols. The projection of $\mathcal{M}_{t+1}$ onto this plane is a rectangle. Figure 3 illustrates the outer ellipsoid obtained during the correction step before applying the decision rule. The feasible strip $\mathcal{S}_{t+1}$ associated with the last output y_{t+1} is such that some candidate solutions present in $\widehat{\mathcal{E}}_{t+1/t}$ are not in $\widehat{\mathcal{E}}_{t+1/t+1}$. The decision rule will now be used to reduce the set of candidate solutions further without having to test individually whether each of the 2^n solutions belongs to $\widehat{\mathcal{E}}_{t+1/t+1}$. Figure 4 illustrates Step 1; the projections of $\widehat{\mathcal{E}}_{t+1/t+1}$ onto the axes associated with the last two symbols make it possible to prove that $d_{t+1} = -1$. Step 2 then implies that $\widehat{\mathcal{E}}_{t+1/t+1}$ should be replaced by its intersection with the plane $d_{t+1} = -1$. On Figure 5, this means replacing $\widehat{\mathcal{E}}_{t+1/t+1}$ by $\widehat{\mathcal{E}}^{\mathbf{new}}_{t+1/t+1}$. Note that other intersections might be needed if $n > 2$, and that all of these intersections can be performed in a single operation. Applying Step 1 again, one can then prove that $d_t = 1$, as illustrated by Figure 6. Other symbols may become known without ambiguity during the process.

Assume that the SNR is independent of t, which amounts to assuming that $\gamma_{t\,\mathbf{max}}$ is constant and can be denoted by $\gamma_{\mathbf{max}}$. Since $\mathbf{c}$ is also constant, the strips $\mathcal{S}_t$ all have the same width and orientation. Figure 7, illustrates the situation for $n = 2$ and $\mathbf{c} = [0.5\ 1]^{\mathrm{T}}$. This impulse response corresponds to a simplified case [8] of a non-minimal-phase channel, which is known to be difficult to equalize with a conventional linear equalizer [11]. Assume, for instance, that the actual values of the symbols are $d_{t-1} = -1$ and $d_t = 1$, again represented by a star on Figure 7. Let $\mathcal{S}_t^+$ and $\mathcal{S}_t^-$ be the strips $\mathcal{S}_t$ for $\gamma_t = \pm\gamma_{\mathbf{max}}$. For any value of γ_t, $\mathcal{S}_t \subset \left(\mathcal{S}_t^+ \cup \mathcal{S}_t^-\right) = \mathcal{U}_t(\mathbf{x}_t)$. Figure 8 presents the strips $\mathcal{U}_t$ for all candidate solutions for $\mathbf{x}_t$ in the case where the SNR is equal to 13 dB. Each $\mathcal{U}_t$ contains only one candidate solution. In such a case the equalization can *always* be made unambiguously, and this is also true for any higher SNR. Figure 9 summarizes the behavior of the algorithm when the value of the previous symbol d_{t-1} is known without ambiguity. The decision part of the algorithm reduces the solution set to a singleton without having to test all the possible solutions, thus estimating d_t unambiguously. This process can be iterated to establish that the values of all symbols can be estimated without error if the value of some initial symbol is known. This reasoning could be extended to the case where $n > 2$ if some sequence of $n-1$ initial symbols is known. Of course, one does not need to assume that these symbols are known to use the algorithm.

Conclusions and perspectives

A bounded-error counterpart to Kalman filtering has been adapted to the equalization of a telecommunication channel. The specific structure of the state equation and the fact that the unknown input and state components all belong to a finite alphabet have been taken into account to obtain a guaranteed equalizer. Guaranteed means here that, provided that the hypotheses of the filter are satisfied, *all* values of the symbols emitted that are consistent with the data received are obtained. Preliminary experimental results, not included because of the lack of space, indicate that the performance of this equalizer is extremely promising. It should be noted that the problem of estimating the impulse response $\mathbf{c}$ of the channel from a known training sequence of symbols can be put in the same framework of state estimation, and treated with the bounded-error tools briefly presented. Further developments are under way to deal with the case where the impulse response of the channel is uncertain or not stationary.

Acknowledgment: The authors thank INTAS for its support under grant INTAS-RFBR-97-10782.

References

1. D.P. Bertsekas and I.B. Rhodes. Recursive state estimation for a set-membership description of uncertainty. *IEEE Trans. on Automatic Control*, AC-16:117–128, 1971.

2. P.L. Combettes. The foundations of set theoretic estimation. *Proc. IEEE*, 81(2):182–208, 1993.

3. J.R. Deller, M. Nayeri and S.F. Odeh. Least-square identification with error bounds for real-time signal processing and control. *Proc. IEEE*, 81(6):813–849, 1993.

4. C. Durieu, B. Polyak and E. Walter. Trace versus determinant in ellipsoidal outer-bounding with application to state estimation. *Proc. 13th IFAC World Congress*, San Francisco, vol. I, 43–48, 1996.

5. — Ellipsoidal state outer-bounding for mimo systems via analytical techniques. *CESA 1996 IMACS*

IEEE-SMC Multiconf., Symp. on Modelling, Analysis and Simulation, Lille, vol. 2, 843–848, 1996.
6. S. Kapoor, S. Nagaraj and Y.-F. Huang. Set-membership adaptive equalization and an updator-shared implementation for multiple channel communications systems. *IEEE Trans. on Signal Processing*, 46(9):2372, 1998.
7. S. Icart, J. LeRoux, L. Pronzato and E. Thierry. Bling equalization in presence of bounded errors. *Proc. Int. Conf. Acoustics, Speech and Signal Processing*, Munich, 3949–3952, 1997.
8. S Marcos, S. Cherif and M. Jaidane. Blind cancellation of intersymbol interference in decision feedback equalizers. *Proc. Int. Conf. Acoustics, Speech and Signal Processing*, Detroit, 1634–1640, 1995.
9. M. Milanese, J. Norton, H. Piet-Lahanier and E. Walter (Eds). *Bounding Approaches to System Identification*. Plenum, New York, 1996.
10. J.P. Norton (Ed.). Special issue on bounded-error estimation. *Int. J. of Adaptative Control and Signal Processing*, 8(1):1–118, 1994 and 9(1):1–132, 1995.
11. J.G. Proakis. *Digital Communications*. Mc Graw-Hill, New York, 1989.
12. F.C. Schweppe. *Uncertain Dynamic Systems*. Prentice Hall, Englewood Cliffs, 1973.
13. E. Walter (Ed.). Special issue on parameter identifications with error bound. *Mathematics and Computers in Simulation*, 32:447–607, 1990.

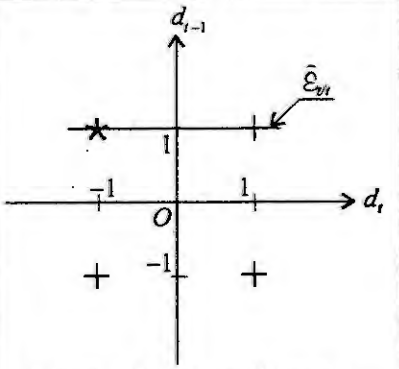


Figure 1: Initial condition at t

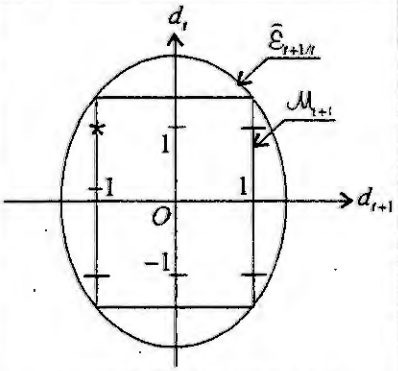


Figure 2: Prediction at $t+1$

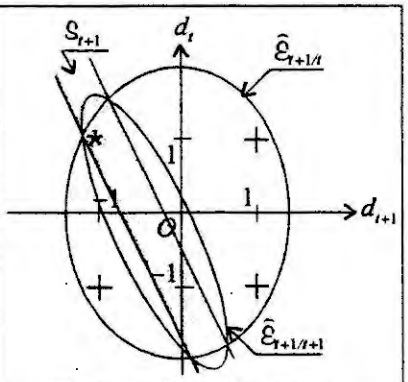


Figure 3: Correction at $t+1$

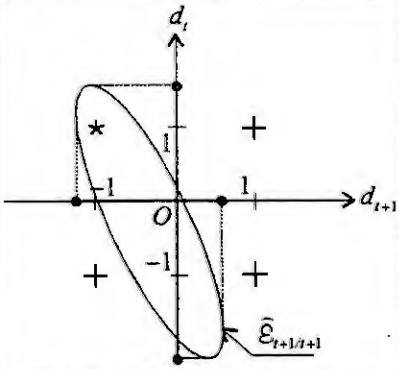


Figure 4: Decision rule Step 1

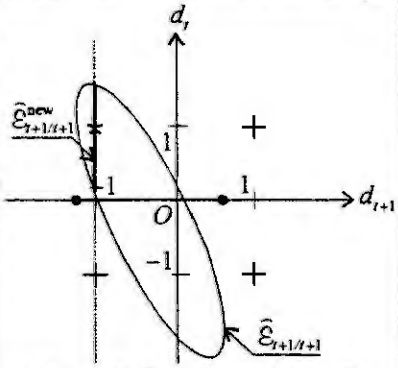


Figure 5: Decision rule Step 2

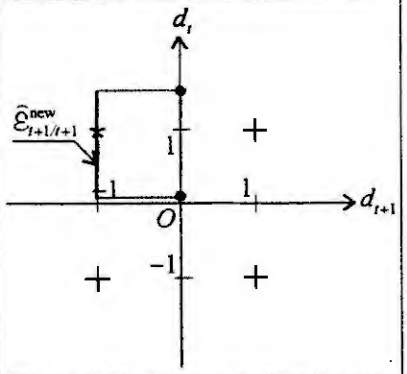


Figure 6: Decision rule Step 1 again

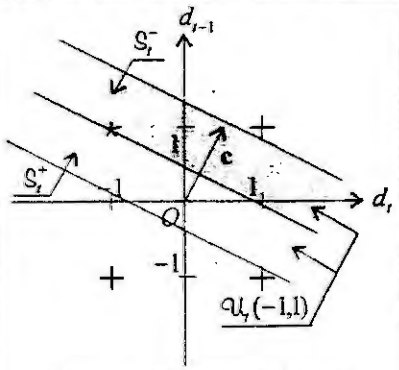


Figure 7: Extreme strips for the symbols emitted ($n=2$)

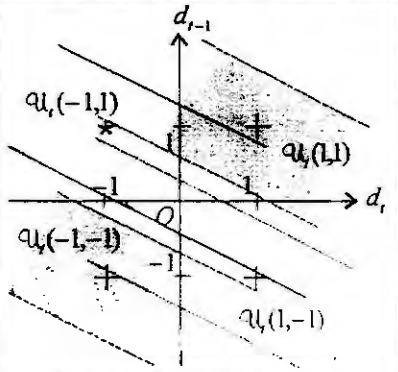


Figure 8: Extreme strips for all candidate solutions ($n=2$)

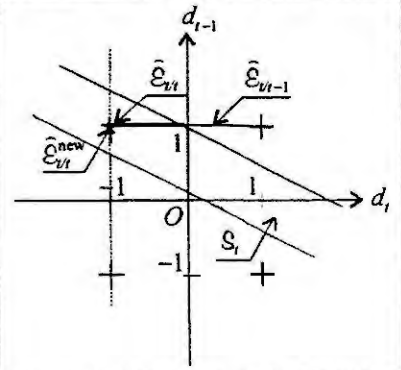


Figure 9: Behaviour of the algorithm when d_{t-1} is known ($n=2$)

REACHABILITY UNDER SET-MEMBERSHIP UNCERTAINTY

[1]A.B.Kurzhanski and [2] P.Varaiya
[1] Moscow State (Lomonosov) University
Fac. of Comput. Math. and Cybernetics, 119899, Moscow, Russia
e-mail: kurzhan@cs.msu.su
[2] University of California at Berkeley
Electrical Eng.@ Comput.Sci, 94720-1770, CA, USA
e-mail: varaiya@eecs.berkeley.edu

Abstract. The report deals with the problem of reachability for systems subjected to uncertain disturbances that are unknown but bounded, with hard bounds. The difference between reachability under closed-loop and open-loop controls is emphasized. It is indicated that the closed-loop reach set may be obtained as the level set for the solution of a "forward" equation of the Hamilton-Jacobi-Bellman-Isaacs type.

1 Introduction

This report is devoted to one of the key issues in control theory – the reachability problem – namely, the computation of the domains reachable by a controlled process through available controls. The reachability problem is usually studied assuming the system model is known. However, in reality the system may always be subject to unknown disturbances and the information on its parameters may not be complete. It then becomes necessary to describe the states reachable by the system *despite the disturbances* or incomplete information or, if exact reachability is impossible, to find the guaranteed errors for reachability. A crucial element in obtaining satifactory reachability properties is that the system has to be governed by *closed – loop controls.* This gives considerable improvement as compared to nonanticipative ("minmax") open-loop controls.

2 The reachability problem

Consider a system of the type

$$\dot{x} = A(t)x + B(t)u + C(t)v(t), \tag{1}$$

with continuous matrix coefficients $A(t), B(t), C(t)$. Here $x \in \mathbb{R}^n$ is the *state* and $u \in \mathbb{R}^p$ is the *control* that may be selected either as an *open loop control* OLC – a Lebesgue-measurable function $u(t)$ of time t, or as a *closed-loop control* CLC – a *set-valued strategy* $\mathcal{U}(t,x)$, restricted respectively by the inclusions

$$u(t) \in \mathcal{P}(t), u = \mathcal{U}(t,x) \subseteq \mathcal{P}(t). \ a.e., \tag{2}$$

Symbol $v \in \mathbb{R}^q$ stands for the unknown *input disturbance* with values $v(t) \in \mathcal{Q}(t)$; $\mathcal{P}(t), \mathcal{Q}(t)$ are set-valued functions with convex compact values, continuous in time.

The class of OLC's $u(\cdot)$ bounded by the first inclusion of (2) is denoted by $U_{\mathcal{O}}$ and the class of boundedc input disturbances $v(\cdot)$ as $V_{\mathcal{O}}$. The strategies $\mathcal{U}$ are taken to be in $U_{\mathcal{C}}$ – the class $U_{\mathcal{C}}$ of CLC's that consists of multivalued maps $\mathcal{U}(t,x)$ bounded by the second inclusion (2), which ensures the existence and extendability of solutions to equation (1), $u = \mathcal{U}(t,x)$, (which now turns into a differential inclusion - DI), for any Lebesgue-measurable function $v(\cdot)$. (This, for example, is the class of set-valued functions with values in $comp\mathbb{R}^n$, upper semicontinuous in x and continuous in t).

Under these conditions equation (1) is said to describe an *uncertain system* of the *linear-convex type.*

Definiton 2.1 *The reach set $\mathcal{X}(\tau, t_0, X^0; \mu)$ of system (1), at time τ, under uncertain input disturbances $v(t)$ is the set of all such points x that satisfy the inclusion $x \in X(\tau, t_0, x^0\ \mathcal{U}) + \mu\mathcal{B}_1(0)$ for some $x^0 \in X^0, \mathcal{U} \in U_C$, $\mu \geq 0$, whatever be the disturbance $v(t) \in \mathcal{Q}(t), t \in [t_0, \tau)$. Here $X(\tau, t_0, x^0\ \mathcal{U})$ is the crossection ("cut") of the solution tube for the differential inclusion*

$$\dot{x} \in A(t)x + B(t)\mathcal{U}(t, x) + C(t)v, \quad x(t_0) = x^0, \tag{3}$$

and $\mathcal{B}_1(0) = \{x : (x, x) \leq 1\}$.

The reachability problem - Problem I, consists in finding the set $\mathcal{X}(\tau, t_0, X^0; 0)$ or, if this set is empty, in finding the set $\mathcal{X}(\tau, t_0, X^0; \mu)$ for some $\mu > 0$. Clearly, for μ sufficiently large, the last set will always be nonempty. It therefore makes sense to look for the smallest of such μ.

3 A related problem of dynamic optimization

. *Problem 2.1*

Find the *value function*

$$\mathcal{V}(\tau, x) = \min_{\mathcal{U}} \max_{x(\cdot)} \{\phi(x(t_0)|x(\tau) = x, \mathcal{U} \in U_C, x(\cdot) \in \mathcal{X}_{\mathcal{U}}(\cdot)\}, \tag{4}$$

where $\phi(x) = d^2(x^0, X^0)$, $d(x, \mathcal{M}) = \min\{(x - z, x - z)^{1/2} | z \in X^0\}$ and $\mathcal{X}_{\mathcal{U}}(\cdot)$ is the variety of all trajectories $x(\cdot)$ of the differential inclusion

$$\dot{x} \in B(t)\mathcal{U}(t, x) + C(t)\mathcal{Q}(t), \ x(\tau) = x, \tag{5}$$

generated by given strategy $\mathcal{U} \in U_C$.

The formal HJBI equation for $\mathcal{V}(t, x)$ is

$$\partial\mathcal{V}/\partial t - \Big(\partial\mathcal{V}/\partial x, A(t)x\Big) - \rho\Big(\partial\mathcal{V}/\partial x | B(t)\mathcal{P}(t)\Big) + \rho\Big(\partial\mathcal{V}/\partial x | C(t)\mathcal{Q}(t)\Big) = 0, \tag{6}$$

with boundary condition

$$\mathcal{V}(t_0, x) = \phi(x(t_0)). \tag{7}$$

Here $\rho(l|\mathcal{Q}) = \max\{(l, q)|q \in \mathcal{Q}\}$ stands for the *support function* of set $\mathcal{Q}$, [4].

Theorem 3.1 *Function $\mathcal{V}(t, x)$ is a generalized ("viscosity") solution of (6, 7).*

Such solutions are explained, for example, in [1]. An important feature is that $\mathcal{V}(\tau, x)$ may be obtained as a limit with $k \to \infty$ of the superposition $V_k^+(\tau, x|t_0, \phi(\cdot)))$ which is defined as follows.

Taking the interval, $T = [t_0, \tau]$, introduce a partition

$$\Sigma_k = \{t_0 = \tau_0, \tau_1, \ldots, \tau_k, \tau = \tau_{k+1}\}, \ \tau_i - \tau_{i-1} = \sigma_i \geq 0, \ i = 1, \ldots, k+1,$$

so that T is now divided into $k+1$ parts

$$T_0 = [t_0, \tau_1), \ T_2 = [\tau_1, \tau_2), \ldots, T_k = [t_1 - \tau_k, \tau],$$

where

$$\tau_i = t_0 + \sum_{j=1}^{i} \sigma_j, \;\; i = 1, \ldots, k,$$

are *the points of corrections.*

Problem 2- k

Solve the following consecutive optimization problems.

Find

$$V_k(\tau_1, x|t_0, \phi(\cdot))) == \min_u \max_v \{\phi(x(t_0)|x(\tau_1) = x, u(t) \in \mathcal{P}(t), v(t) \in \mathcal{Q}(t), t \in T_1\},$$

where $\phi(x) = d^2(x^0, X^0)$, then consecutively, for $i = 2, \ldots, k$, find

$$V_k(\tau_i, x|\tau_{i-1}, V_{i-1}(\tau_{i-1}, x(\tau_{i-1})|\cdot)) =$$

$$= \min_u \max_v \{V_k(\tau_{i-1}, x(\tau_{i-1})|\cdot)|x(\tau_i) = x, u(t) \in \mathcal{P}(t), v(t) \in \mathcal{Q}(t), t \in T_i\},$$

and finally

$$V_k(\tau, x|V_k(\tau_k, x(\tau_k)|\cdot)) =$$

$$\min_u \max_v \{V_k(\tau_k, x(\tau_k)|\cdot)|x(\tau) = x, u(t) \in \mathcal{P}(t), v(t) \in \mathcal{Q}(t), t \in T_{k+1}\}.$$

so that

$$V_k(\tau, x|t_0, \phi(\cdot)) = \tag{8}$$

$$V_k(\tau, x|\tau_k, V_k(\tau_k, x(\tau_k)|\ldots|\tau, V_k^-(\tau_1, x(\tau_1)|t_0, \phi(\cdot))\ldots)$$

The convergence theorem sounds as follows.

Theorem 3.2 *Suppose functions $V_k(\tau, x)$ are proper convex. Consider a sequence of partitions Σ_k with $k \to \infty$ and*

$$\max\{\sigma_i \; : \; i = 1, \ldots, k+1\} \to 0, \; \sum_{i=1}^{k+1} \sigma_i = \tau - t_0. \tag{9}$$

(Without loss of generality it may be always considered monotone, namely, achieved by adding new points of correction to the old ones).

Then there exists a pointwise limit

$$\lim_{k \to \infty} V_k(\tau, x) = \mathcal{V}_0(\tau, x). \tag{10}$$

The limit do not depend on the type of partititons Σ_k and coincides with $\mathcal{V}(\tau, x)$, namely

$$\mathcal{V}_0(\tau, x) = \mathcal{V}(\tau, x).$$

In this case there also holds the property of *epiconvergence* of the epigraph of $V_k(\tau, x)$ to the epigraph of $\mathcal{V}_0(\tau, x)$ (see [4]).

4 The reach set under uncertainty

Theorem 4.1 *The closed-loop reach set of level μ (under complete feedback) is the level set*

$$\mathcal{X}(\tau, t_0; \mu) = \{x : \mathcal{V}(\tau, x) \leq \mu)\}. \tag{11}$$

Here we note that the following inequalities are true:

$$\mathcal{V}(\tau, x) \leq \ldots \leq V_k(\tau, x) \leq \ldots \leq V_0(\tau, x).$$

Therefore, the level set

$$\mathcal{X}_k(\tau, t_0 : \mu) \subseteq \mathcal{X}(\tau, t_o, \mu), \ \forall \ \mu \geq 0.$$

whatever be the integer $k \geq 0$.

We note that

$$\mathcal{X}_k(\tau, t_0; 0) = \{x : V_0(\tau, x) \leq 0\}$$

where

$$V_0(\tau, x) = \min_u \max_v \{d^2(x(t_0), X^0) | u(t) \in \mathcal{P}(t), v(t) \in \mathcal{Q}(t), \ t \in [t_0, \tau]\}$$

is the solution to the nonanticipative (minmax) problem in the class $\mathcal{U}_{\mathcal{O}}$ of open-loop controls.

Conclusion

In this paper we introduce the notion of reach sets under feedback (closed – loop) control and set-membership input uncertainty, with hard bounds on the unknowns. We indicate that these sets may be obtained as the levels sets of the viscosity solution to a *forward* HJBI (Hamilton-Jacobi-Bellman-Isaacs) equation. These sets naturally turn to be larger than the respective sets for a similar nonanticipative minmax problem of open–loop control.

References

[1] FLEMING W.H., SONER H.M., Controlled Markov Processes and Viscosity Solutions, Springer - Verlag, 1993.

[2] KRASOVSKI N.N., SUBBOTIN A.N., Positional Differential Games, Springer-Verlag, 1988.

[3] KURZHANSKI A.B. Pontryagin's alternated integral in the theory of control synthesis. Proc. Steklov Math. Inst., v.224, 1999, (Engl. Transl.), pp. 234 - 248.

[4] ROCKAFELLAR R.T., WETS R.J.B., Variational Analysis, Springer-Verlag, 1998.

[5] VARAIYA P., LIN J., Existence of Saddle Points in Differential Games, SIAM Journal on Control and Optimization, v.7,1, 1969, pp.142 - 157.

GRADED SET-MEMBERSHIP MODELS

J. P. Norton[1] and P. F. Weston
School of Electronic & Electrical Engineering, University of Birmingham,
Edgbaston, Birmingham, B15 2TT, U. K.
[1]author for correspondence: email j.p.norton@bham.ac.uk

Abstract. The parameters or state variables of a set-membership model are classified only as feasible (consistent with the specified error bounds and observations) or not. While compatible with worst-case design or prediction, this is uninformative about quality of fit to the observations and neglects such distributional information as is usually available. The paper addresses the problems of grading the feasible set according to how well each feasible value matches the data, or whether it meets crude requirements on distribution. Computing load and approximation are considered.

Introduction

Set-membership models specify the uncertainty in forcing, observation error and, where applicable, the initial state or parameters, by bounds. The observations are processed so as to categorize all values of state variables or parameters only as feasible or not, with no order of preference. While this is compatible with worst-case design or prediction and avoids the need to specify any distributional properties, it is uninformative about how well the model fits the data and it neglects distributional information which is usually available, even if partial and approximate. Such information may result from knowledge of the mechanism generating model-output error or forcing, but may alternatively be the means of stating the performance required of the model; for example, a certain small proportion of observations may be treatable as bad data, not required to give model-output errors within the usual bounds, or one may be prepared to tolerate poor model performance for some part of the time.

The paper therefore discusses the problems of grading feasible values by how well they match the observations or collateral knowledge on distribution. To cover both parameter and state estimation, models linear in the unknowns θ, of the form

$$\left.\begin{aligned} y_k &= \phi_k^T \theta_k + v_k \\ \theta_{k+1} &= F\theta_k + Gw_k \end{aligned}\right\} \quad k = 1,2,..,N \tag{1}$$

with noise/forcing bounds

$$|v_k| \le \varepsilon_v, \quad |w_k| \le \varepsilon_w \tag{2}$$

are considered. For bounding of constant parameters, F is identity and ε_w is zero.

It is relatively easy to devise grading schemes with no regard to computational loads, but harder to find computationally economical schemes. For grading by fit to the observations, possibilities include various norms of the vector of successive model-output errors (with (2) constraining its l_∞ norm), measures of distance from the boundary of the feasible set, and measures of how far ε_v and ε_w would have to be tightened to render the value unfeasible. Of these, the first is very easy in the l_2 case, where the constant-norm contours are readily computed, and fairly easy in the l_1 case. The minimum distance to the boundary of the feasible set is easily computed for piecewise linearly bounded (polytope) sets or their ellipsoidal approximations, and contours of constant distance are equally easily constructed. The third measure, maximum amount of tightening ε_v or ε_w, can be computed for fixed-parameter models by intersecting the half spaces defined by hyperplane bounds

$$\phi_k^T\theta - \varepsilon_v \le y_k, \quad y_k \le \phi_k^T\theta + \varepsilon_v \tag{3}$$

in the space of (θ, ε_v), the parameters and observation-noise bound jointly. The active hyperplane bounds define a polyhedron, unbounded in the direction of increasing ε_v, which directly gives the minimum ε_v for which any given θ remains feasible. For state-bounding (or bounding of time-varying parameters treated as state variables), the problem is harder to visualise, as the time- and observation-updating steps are in different subspaces of $(\theta_k, \varepsilon_v, \varepsilon_w)$. However, the influence of varying each noise bound is linear, and the paper discusses how far a similar approach suffices.

The second problem, grading feasible values according to how they meet distributional requirements, has two aspects: identifying the restricted set of values which as well as being feasible with respect to the overall (outermost) bounds yield an error sequence with an acceptable distribution, and for those which do not, determining by what margin they fail. The former has been considered for a distributional specification consisting of bounds on the contents of a number of histogram bins [1] but is computationally heavy [2]. The

measurement of the discrepancy, for the feasible-overall but ill-distributed values, should depend on the application of the bounds of θ_k, as assessment of the seriousness of too many or too few errors being in certain ranges relies on insight into the consequences of various types of model misfit.

The main aims of the paper are to suggest some useful grading schemes, note which look computationally acceptable, and see how standard or near-standard operations on bounds can contribute to grading. Examples illustrate the results obtainable and the difficulties arising.

Grading

Let the process noise $\{w_k\}$ be zero for now, so that the feasible set is a polytope defined by hyperplane bounds due to $|v_k| \le \varepsilon_v$. If a feasible point is graded on its distance from each active hyperplane in turn, there are two weaknesses. First, the distance is $(\varepsilon_v \pm v_k(\theta))/\|\phi_k\|_2$, but the scaling by $\|\phi_k\|$ is easily taken into account. Second, inactive hyperplanes and the corresponding output errors are ignored, so grading measures how far the output error is from $\pm\varepsilon_v$ only for the subset of observations yielding active bounds on θ. The subset changes with ε_v.

Now consider the effect of process noise. The state update comprises linear transformation by F, which does not change the active bounds, then vector summing with the set of feasible forcing Gw_k. Commonly, the set $\mathcal{W}_k$ of possible Gw_k is the vector sum of vectors (one per independent forcing source). The effect of adding an extreme of Gw_k to the feasible half space defined by any active (hyperplane) bound is an easily computed translation. The vector sum of the half space and $\mathcal{W}_k$ is thus found by translating the hyperplane by the extreme of $\mathcal{W}_k$ in the direction of the outward normal to the hyperplane. If this is done for all hyperplane bounds, active and inactive, the resulting intersection of $2k$ half spaces differs from the vector sum of $\mathcal{W}_k$ and the F-transformed feasible set for θ only by exceeding it near vertices defined by hyperplanes with translated positions determined by different extrema of $\mathcal{W}_k$. This approximation keeps the computing load linear in the number of observations (fixed, if a window is employed) and might be improved by identifying and rectifying those vertices where it is poor.

Moreover, the effect of uncertainty in process noise scaling is to define hyperplane pairs in the form $\alpha_{1,k} - \beta_k\lambda \le \varphi_k^T\theta_k \le \alpha_{2,k} + \beta_k\lambda$ where $\alpha_{1,k}$, $\alpha_{2,k} \ge \alpha_{1,k}$ and $\beta_k \ge 0$ are readily calculated constants and $\lambda \ge 0$ scales the process noise. If, also, the observation noise is scaled by $\eta \ge 0$, the result is of the form $\alpha_{1,k} - \beta_k\lambda - \gamma_k\eta \le \varphi_k^T\theta_k \le \alpha_{2,k} + \beta_k\lambda + \gamma_k\eta$ where $\gamma_k \ge 0$ is a further constant. Thus the motion of the hyperplanes is linear in the scaling of the process and observation noises. For any candidate θ_k, one can determine that each hyperplane will render θ_k infeasible for certain scalings of the observation and process noise, leading to a linear relationship between λ and η for each point. In this way the scaling of the noises for θ_k to just remain feasible can be computed for any desired relative scaling λ/η. The feasible set is thus graded by robustness to scaling of the process and observation noises. The approximation introduced above means that the feasible set for a given choice of scale contains the true feasible set, with extra volume introduced near some vertices. The advantage is a large saving in computation. Of particular interest might be the θ_k most robust to noise scaling, i.e. defining the tightest attainable noise specifications.

Grading Feasible Values According to Distributional Requirements

If there is no process noise, each feasible parameter value defines a sequence of observation errors whose distribution is immediately testable via some criterion such as mean squared error. Despite this, the computational effort is high unless the noise distribution can be transformed analytically into a p.d.f. in the parameter space (e.g. Gaussian distributions). When there is process noise in the system, unless the distribution of the observation and process noises are Gaussian, distributional information is difficult to obtain. This is because many trajectories can lead to the same final state, so the distribution is difficult to test for a feasible state value. To do so, one must solve an optimisation problem for each candidate point, clearly not practicable because of the convolution in each time-update step.

The bounds $|v_k| \le \varepsilon_v$ allow all observation errors to be extremal. The observation errors may be constrained more by constraining their sum of squares. In the parameter-estimation case,

$$\frac{1}{N}\sum_{k=1}^{N}\left(y_k-\phi_k^{\mathrm{T}}\theta\right)^{\mathrm{T}}\left(y_k-\phi_k^{\mathrm{T}}\theta\right)\le\sigma^2 \tag{4}$$

This defines an ellipsoidal region in θ space, of the form

$$\theta^{\mathrm{T}}\left(\sum_{k=1}^{N}\phi_k\phi_k^{\mathrm{T}}\right)\theta-2\left(\sum_{k=1}^{N}y_k\phi_k^{\mathrm{T}}\right)\theta+\sum_{k=1}^{N}\left(y_k^{\mathrm{T}}y_k-\sigma^2\right)\le 0 \tag{5}$$

As an example, 10 scalar observations were made of a vector of two parameters with a true value of $\begin{bmatrix}0 & 0\end{bmatrix}^{\mathrm{T}}$, indicated by a square in Fig. 1. The directions of the ϕ_k were chosen roughly uniformly with magnitudes between 0.7 and 1, the observation errors were $\{0.5,-0.3,-0.7,0.1,0.8,0.2,-0.9,0.4,0.1,-0.7\}$. The model had $\varepsilon_v=1$. Fig. 1 shows the observation hyperplanes and the polytope feasible region. A bound on $\frac{1}{N}\sum_{k=1}^{N}v_k^2$ was varied from 0 to 1.0 to obtain a set of outer bounding ellipsoids. The true $\frac{1}{N}\sum_{k=1}^{N}v_k^2$ is 0.299. The $\{v_k\}$ might be considered as from a distribution where 6 observation error magnitudes between 0 and 0.5, and 4 between 0.5 and 1. The worst case variance for such a distribution is 0.55. The ellipsoid defined by such a constraint on the observation-error variance is indicated by the thickest ellipsoid in Fig. 1. The ellipsoid contains the feasible region and thus adds no useful information. For smaller σ^2 the concentric ellipsoids collapse to zero volume giving a lower bound on σ^2 of 0.262 at $\theta=\begin{bmatrix}-0.3642 & -0.1029\end{bmatrix}$.

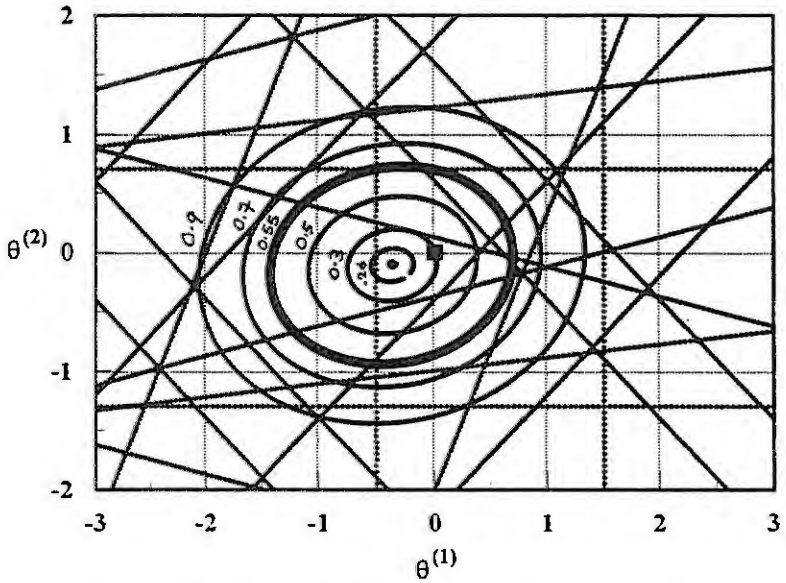


Fig. 1: Feasible Set and Constant Variance Ellipsoids

What does this show? In this example, the variance of the observation errors is large relative to the observation-error bound. There is a high probability of the observation error being near extreme, so the feasible set is well defined (at least by the polytope solution). Rough distributional information yielded no additional information in this example. Had it indicated that the variance was restricted enough to improve the feasible set defined by the hyperplane bounds, they would very likely have been further away. In that case, the variance constraint may well lie within all or most active bounds. So if one considers extreme observation errors unlikely, the feasible set from active bounds is likely to be loose and distributional information useful. However, if observation errors are often large, loose distributional information is dominated by the active bounds. This suggests that bounds and other distributional information will work well in combination only if well balanced. A minimum volume outer bounding ellipsoid for this case (by batch calculation using an LMI (linear matrix inequality) formulation of the problem [3]) is outside most of the contours of constant sum of squares of errors. Therefore, the outer bounding ellipsoid provides no information to supplement a bound on sum of squares of errors.

Modifying a Limited Number of Bounds

In the previous section, the feasible set was graded by the observation-error variance. It is defined by active bounds that correspond to extreme observation errors. If those bounds are set high to accommodate occasional large errors, then variance information is likely to define feasibility better than the active bounds due to extreme errors, implying that distributional information is more useful. However, variance constraints are strongly influenced by large observation errors, even if these are relatively infrequent. Therefore, consider the following scheme designed to give a choice of feasible sets conditioned on the number of errors within wide error bounds, with most errors between narrower bounds.

Let there be M feasible sets $\Omega_{k,m}$, $m = 1,\cdots,M$ after observation k has been processed, where set m corresponds to m observations being between the wide bounds and the rest between the narrower ones. Let observation y_k define two sets for θ, Ω_k^n and Ω_k^w for the narrower and wider error bounds respectively.

$$\Omega_{k,m} = \left(\Omega_{k-1,m} \cap \Omega_k^n\right) \cup \left(\Omega_{k-1,m-1} \cap \Omega_k^w\right), \quad m = 1,\cdots,M \tag{6}$$

where $\Omega_{k-1,-1}$ is treated as the empty set. The state update is unchanged from the normal case, but one has to process M regions instead of one. N observations give M sets (some perhaps empty) indicating that the data are inconsistent with fewer than $m = 1,2,...,M$ observation errors within the wider bounds. Decisions are then based on this number. One can hedge by examining the effect of choosing too few large observation errors and risking a feasible set not containing the true state.

A difficulty with this update scheme is that the union operation in (6) typically results in a non-convex set. To represent the feasible sets as polytopes or ellipsoids, one has to approximate the union and intersection operations, for example using minimum-volume ellipsoids with recursive operations or faxed-lag batch calculations (one can write down batch ellipsoidal problems as LMI problems that can be efficiently solved [3]).

This process is easily extended to narrow and wide bounds for process noise, as long as one need not distinguish between process or observation errors within the wide bounds. One can also handle division of the observation and/or process noises into more than two ranges, but the number of combinations soon becomes too large. Where one expects one or two errors within wider bounds (unbounded if desired), this method gives rise to three feasible sets: with all errors in the narrow range, with one error in the wider range and with two errors in the wider bounds. One interesting effect of approximating the intersection and union operations is that the region allowing m errors in the wider bounds can extend beyond the region allowing $m+1$ (or more) errors in the wider bounds. In this case, one can improve the approximation of $\Omega_{k,m}$ by excluding the regions $\Omega_{k,m+l}$ with positive l.

If the data cannot be explained with fewer than j larger errors, then some of the feasible sets may disappear (although with the approximations some infeasible sets still appear feasible). Thus the number of feasible sets to be maintained need not remain at M. Overlapping feasible sets in θ space indicates the variation in the feasible set with different problem specifications, in terms of number of errors within the wider bounds. An example will be presented at the conference.

Conclusions

This paper has discussed some ways in which a feasible set can be augmented with additional information. To obtain a tight feasible set, some model output-errors must be close to their error bounds. If those bounds are loose, so is the feasible set. A bound on the variance of the observation errors can lead to a tighter feasible set than do observation-error bounds alone. By maintaining multiple feasible sets, one can identify feasible sets consistent with differing proportions of small and large errors. Some of the feasible sets disappear, ruling out certain model specifications; some models will be feasible but actually incorrect, although consistent with the data; a third set of feasible models represents the truth. Interpretation must therefore take the purpose of the feasible set into account.

References

[1]. J. P. Norton, Distributional bounding models of uncertain systems, Preprints SYSID '94, 10th IFAC Symp. On System Identification, Copenhagen, (1994), 2, 91-95.

[2]. A. J. Downing and J. P. Norton, Computation of parameter bounds from bounded-histogram error specifications, Preprints IFAC World Congress, San Francisco (1996), I, 55-60.

[3]. Vandenberghe, L., Boyd, S., and Wu, S.-P., Determinant maximization with linear matrix inequality constraints, Tech. Report, Info. Systems Lab., Dept Elec. Eng., Stanford University, CA 94305, (1996).

ELLIPSOIDAL STATE ESTIMATION OF PERTURBED LINEAR SYSTEMS IN THE PRESENCE OF OBSERVATION ERRORS

A.N.Kinev, D.Ya.Rokityanskii, F.L.Chernousko
Institute for Problems in Mechanics of the Russian Academy of Sciences
pr. Vernadskogo 101-1, 117526 Moscow, RUSSIA

Abstract. Linear dynamic systems described by finite-difference are considered. The matrix of the system is uncertain or subject to disturbance, and only the bounds on admissible perturbations of the matrix are known. Outer ellipsoidal estimates on reachable sets of the system are obtained using the method of ellipsoids and equations described the evolution of the approximating ellipsoids are derived.

Introduction

Dynamic systems with unknown, uncertain, or perturbed parameters, in the presence of observation errors of various nature appear in numerous applications. In such systems it is important to obtain bounds on reachable sets, i.e., bounds on all possible motions of these systems when they are subjected to unknown bounded perturbations, and some data on discrete or continuous observations are available. In the paper, linear systems described by ordinary finite-difference are considered. The matrices describing the dynamics of the system and the observation process are assumed to be uncertain: they belong to known subsets of the spaces of matrices of corresponding dimensions. These systems can serve as models for various mechanical, electrical, and other systems whose parameters are not known precisely or can vary in an uncertain way. Outer ellipsoidal estimates on reachable sets are obtained using the method of ellipsoids. Ellipsoidal estimates have a number of advantages such as simple and explicit form of approximation, smooth boundaries, invariance with respect to linear transformations, etc. [1-4].

Example of two-dimensional system with a perturbed matrix and discrete observations with bounded errors is presented.

Statement of the problem

Discrete-time system described by linear finite-difference equations

$$x(t_{i+1}) = C(t_i)x(t_i) + f(t_i), \quad t_0 < t_1, ..., \quad i = 0, 1, ... \tag{1}$$

is considered in this section. Here, $f \in R^n$, $x \in R^n$ is the state vector, and the matrix C belongs to a set of $n \times n$ real-valued matrices M_n. All vectors and matrices $x(t_i)$ and $C(t_i)$ are defined at given discrete instants of time t_i, $i = 0, 1, ...$, the vector function $f(t_i)$ is a given function of time, and the matrix $C(t_i)$ is represented as the sum

$$C(t_i) = C_0(t_i) + C_1(t_i), \quad i = 0, 1, ... \tag{2}$$

Here, $C_0(t_i)$ is a given non-singular matrix depending on time, and C_1 is unknown. The matrix C_1 contains parametric uncertainties, and we assume it to be subjected to the constraints [5]

$$C_1(t_i) \in V(t_i) \subset M_n, \quad i = 0, 1, ... \tag{3}$$

We assume that the set $V(t_i)$ of all possible perturbation matrices is a ball of radius h in M_n defined by

$$V = \{C_1 \in M_n : \ ||C_1||_{l_p} \le h(t_i)\}, \tag{4}$$

where the function h is given at time instances t_i and the norm $||C_1||_{l_p}$ is defined as follows: $(1 \le p \le \infty)$:

$$||C_1||_{l_p} = \left(\sum_{1 \le i,j \le n} |c_{ij}|^p \right)^{1/p}, \quad ||C_1||_{l_\infty} = \max_{1 \le i,j \le n} |c_{ij}|. \tag{5}$$

Other cases of defining the set V were also considered, including one providing bounds on each element of the matrix C_1 (see example below) [5, 6].

The initial state of system (1) is unknown. We assume that the set M containing the initial state is given:

$$x(t_0) \in M, \quad M \subset R^n. \tag{6}$$

Let $z \in R^m$ be a vector of the first m components of the state vector x. We assume that, at each t_i, the results of observations $y(t_i) \in R^m$ are available for $i = 0, 1, ...$, where

$$y(t_i) = B(t_i)z(t_i) + w(t_i). \tag{7}$$

Here, measurement errors are represented by vector $w(t_i) \in R^m$ and matrix $B(t_i) \in M_m$. Matrix $B(t_i)$ is represented as the sum

$$B(t_i) = B_0(t_i) + B_1(t_i), \quad i = 0, 1, \ldots . \tag{8}$$

Here, $B_0(t_i)$ is known and non-singular, and $B_1(t_i)$ is unknown. We assume the following constraints to be imposed

$$B_1(t_i) \in F(t_i) \subset M_m, \quad i = 0, 1, \ldots . \tag{9}$$

Here, $F(t_i)$ is assumed to be a ball of radius $b(t_i)$ in M_m defined by

$$F(t_i) = \{B_1 \in M_m; \quad ||B_1||_{l_p} \leq b(t_i)\}. \tag{10}$$

For w the following inequality hold:

$$|w|_p \leq g(t_i), \quad 1 \leq p \leq \infty$$

$$|w|_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p}, \quad |w|_\infty = \max_{1 \leq i \leq n} |x_i|, \tag{11}$$

In (10) and (11) functions b and g are given and reasonably small (for all constructed estimates to be effective the following inequality should hold for any t_i: $b(t_i) \leq ||B_0^{-1}(t_i)||_p$).

For other cases of defining the set V by equations (4) and (5) containing possible perturbation matrices C_1, equations (10) and (11) are also changing (see example).

Denote by $D_x(t_i)$ the set of all possible values of the state vector $x(t_i)$ (satisfying equations(1), all constraints imposed, and initial conditions (6)) which are compatible with (do not contradict) the observation results $y(t_j)$ for $j \leq i$.

We introduce the notation $E(a, Q)$ for an n-dimensional ellipsoid

$$E(a, Q) = \{x : \ (Q^{-1}(x - a), (x - a)) \leq 1\}, \tag{12}$$

There arises the problem of the outer ellipsoidal approximation of the set $D_x(t_i)$ (similar to the one for reachable sets in case when observations are absent, see [3]). It can be formulated as follows.

Problem 1. Find a vector-valued function $a(t_i)$ and a matrix-valued function $Q(t_i)$ such that

$$D_x(t_i) \subset E(a(t_i), Q(t_i)), \quad i = 0, 1, \ldots . \tag{13}$$

The advantages of the ellipsoidal approximations include relative simplicity, smoothness of the boundary, and invariance with respect to linear transformations, etc.[2, 3].

A similar problem for a continuous-time formulation for either continuous or discrete observations can be stated and solved using the same approach.

Problem 1 has many solutions: any ellipsoid containing a solution is also a solution. Therefore, it is natural to ask whether the solution can be minimized in the sense of some optimality criterion characterizing the "size" of the approximating ellipsoids, for example their volume, sum of squared semi-axes, etc. This approach was adopted previously in [2], [3] for additive perturbations. In this paper, approximating ellipsoids possessing the minimal volume property will be obtained.

Outer Approximation for Discrete-Time Systems

Summing up the results, we can present the solution of Problem 1. To obtain it, we shall describe the procedure of constructing the functions $a(t_i)$, $Q(t_i)$ occurring in (13). We choose an ellipsoid $E(a_0, Q_0)$ containing the initial set M in (6) (for example, an ellipsoid of the minimum volume) so that $M \subset E(a_0, Q_0)$ and set

$$a(t_0) = a_0, \quad Q(t_0) = Q_0. \tag{14}$$

The vector $a(t_{i+1})$ and the matrix $Q(t_{i+1})$ are recursively expressed in terms of $a(t_i)$, $Q(t_i)$, and known functions of t_i, $i = 0, 1, \ldots$. Let us denote:

$$R_p(t_i) = \max_{x \in E(a(t_i), Q(t_i))} |x|_p. \tag{15}$$

We shall introduce the following notation

$$Q_1 = C_0(t_i) Q(t_i) C_0^T(t_i), \tag{16}$$

$$Q_2 = h(t_i) R_p(t_i) n^{1/p - 1/2} I, \tag{17}$$

where I is the identity matrix. Let the numbers $\lambda_j \geq 0$, $j = 1, \ldots, n$, be the roots of the characteristic equation

$$\det(Q_1 - \lambda Q_2) = 0, \tag{18}$$

each root being counted according to its multiplicity. Then we define $\varkappa > 0$ as the unique positive root of the equation

$$\sum_{j=1}^{n} \frac{1}{\varkappa + \lambda_j} = \frac{n}{\varkappa(\varkappa + 1)}. \tag{19}$$

We shall denote variables $a^+ \in R^n$ and $Q^+ \in M_n$:

$$a^+ = C_0(t_i) a(t_i) + f(t_i), \tag{20}$$

$$Q^+ = (\varkappa^{-1} + 1) Q_1 + (\varkappa + 1) Q_2,$$

Now we can state that here exists a unique minimum volume ellipsoid which yields a solution of Problem 1. This ellipsoid is the optimal one containing the convex set which is the intersection of the ellipsoid $E(a^+, Q^+)$ and the set U which is based on the observatior results and is given by:

$$U = \left\{ x : (x_1, \ldots, x_m) = z,\ |z - B_0^{-1} y(t_i)|_p \leq \frac{||B_0^{-1}(t_i)||_{l_p} b(t_i)(|y(t_i)|_p + g(t_i))}{1 - ||B_0^{-1}(t_i)||_{l_p} b(t_i)} \right\}$$

Similar results (in the form of ordinary differential evolution equations for the center of the ellipsoid $a(t)$ and the matrix $Q(t)$) are obtained for approximation of the set $D_x(t)$ for continuous-time systems with observations subjected to additive and multiplicative errors.

Example

We consider the two-dimensional system

$$\dot{x}_1 = x_2, \quad \dot{x}_2 = -x_1 + c(t) x_1, \quad |c(t)| \leq N, \tag{21}$$

where $c(t)$ is an unknown bounded perturbation, and N is a positive constant. If $c(t)$ is a periodic function, system (21) describes parametric excitation of oscillations.

Discrete observations (with unknown but bounded errors) are assumed to be available. A two dimensional vector $y = \{y_1, y_2\}$ is observed:

$$y_i(kT) = (1 + p_i(kT)) x_i(kT) + w_i(kT), \quad i = 1, 2, \quad k = 0, 1, \ldots \tag{22}$$

Here $p_i \leq m_i, i = 1,\ 2,\ w_i \leq M_i, i = 1,\ 2$

Suppose that the initial set M at time $t = 0$ is a disk of unique radius in the $x_1 x_2$-plane centered at the origin. Then

$$a_1(0) = a_2(0) = 0,$$

$$Q_{11}(0) = Q_{22}(0) = 1, \quad Q_{12}(0) = 0. \tag{23}$$

The results of the numerical simulation are shown in Fig. 1. Observations (22) with $M_1 = 1$, $M_2 = 1$, $m_1 = 0.1$, $m_2 = 0.1$ available at discrete time instances $t = 0, 1, \ldots$ $(T = 1)$ were used to correct the matrix and center of the ellipse. The function $c(t)$ was defined as follows: $c(t) = \operatorname{sign} \sin(5t)$. Recall

that the support function of the ellipse $E(a, Q(t))$ is $H_E(z) = a + (Qz, z)^{1/2}$. Therefore, the quantities $a_i - Q_{ii}^{1/2}$ and $a_i + Q_{ii}^{1/2}$ provide the lower and upper bounds, respectively, on the projections of the ellipse $E(a, Q(t))$ onto the axes x_i, $i = 1, 2$. To illustrate the effectiveness of the method proposed in the paper, the curve of the trajectory (solid one) is presented along with the lower and upper bounds (dashed curves) corresponding the constructed ellipsoidal estimates.

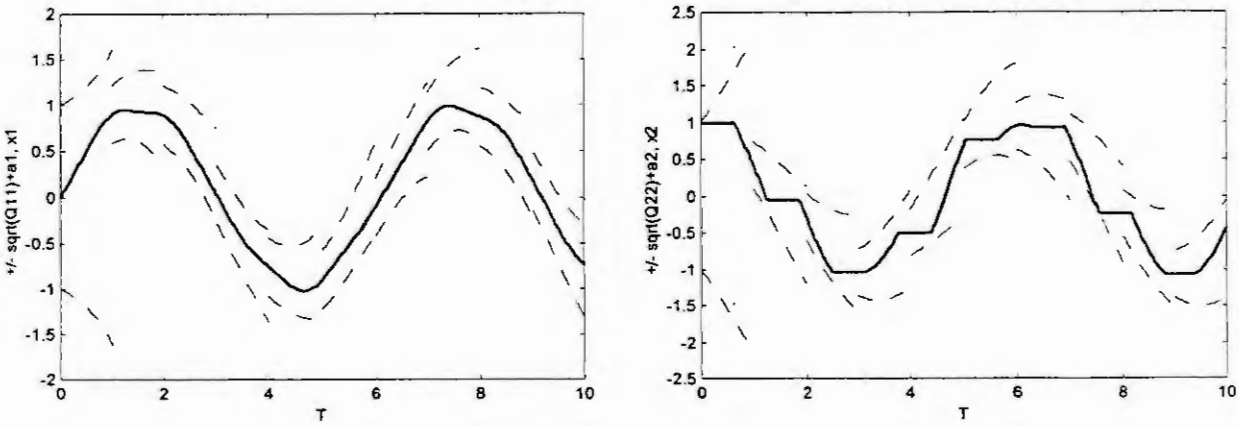


Fig. 1. Simulation results.

Conclusions

The evolution equations for outer ellipsoidal bounds on sets $D_x(t_i)$ which are compatible with both equations of motion and provided observations are obtained in this paper. These bounds can be useful for estimating the influence of parametric uncertainties and perturbations affecting the matrices of linear dynamical systems with observation errors.

Acknowledgments. This work was supported by the Russian Foundation for Basic Research (Grant 99-01-00258) and by International Association for the promotion of cooperation with scientists from the New Independent States of the former Soviet Union (Grant INTAS-RFBR 97-10782).

References

1. Schweppe F.C., Uncertain Dynamic Systems, Prentice Hall, Englewood Cliffs, USA, 1973.
2. Chernousko, F.L., Optimal guaranteed estimates of indeterminacies using ellipsoids, Izvestiya Akademii Nauk SSSR, Tekhnicheskaya Kibernetica, Vol.I, No.3, pp. 3-11; Vol.II, №. 4, pp. 3-11; Vol. III, No. 5, pp.5-11, 1980.
3. Chernousko F.L., State Estimation for Dynamic Systems, CRC Press, Boca Raton, Florida, USA, 1994.
4. Kurzhanski A. B., Valyi I., Ellipsoidal Calculus for Estimation and Control, Birkhäuser, 1996.
5. Chernousko F.L., Estimation of the attainability sets of linear systems with an indeterminate matrix, Doclady Mathematics, Vol. 54, No 1, pp. 634-636, 1996.
6. Rokityanskii D.Ya., *Perturbed Linear Mappings,* Journal of Computer and Systems Sciences International, No 1, pp. 110-117, 1997

ON MODELLING OF CONTROLS AND UNCERTAIN DISTURBANCES IN DYNAMICAL SYSTEMS

F.L.Chernousko
Institute for Problems in Mechanics of the Russian Academy of Sciences
pr. Vernadskogo 101-1, 117526 Moscow, RUSSIA

Abstract. Optimal controls for a simple dynamical system with one degree of freedom are presented and compared under various sets of assumptions and constraints imposed on the control and disturbance.

Introduction

For the control design and the analysis of behavior of dynamical systems subjected to uncertain disturbances, it is important to specify adequately the control constraints and the class of possible disturbances. Inadequate assumptions may lead to oversimplified models and result in controls which cannot be implemented in practice. For example, bang-bang optimal controls based on the only assumption that the control is bounded are not always realizable. We present and compare optimal and suboptimal controls for a simple dynamical system with one degree of freedom

$$\ddot{x} = u + v \tag{1}$$

under various sets of assumptions and constraints imposed on the control u and disturbance v. Optimal control problems both with fixed and free time interval are considered. The obtained results are based on the theories of optimal control and differential games [2,3]. The calculations are omitted and the attention is given to the problem statements and the comparative presentation of the final results.

Optimal control over a fixed time interval

Suppose the time interval is fixed: $t \in [0, T]$, and the control u is to minimize the terminal cost functional $J = |x(T)|$ under the zero initial conditions: $x(0) = \dot{x}(0) = 0$. By normalizing time, we can assume without loss of generality that $T = 1$. Then $J = |x(1)|$.

We consider six sets of conditions imposed on the control u and disturbance v.

1. Both u and v are bounded by given constants:

$$|u(t)| \le u, \qquad |v(t)| \le v \tag{2}$$

2. The control u is an impulse, whereas v is bounded:

$$u(t) = U\delta(t - \tau), \qquad |U| \le u^0, \qquad \tau \in [0, 1], \qquad |v(t)| \le v^0$$

Here δ is the delta-function, and the constants U and τ can be chosen in the optimal way by the first player (u) who minimizes J.

3. The control u is bounded, whereas v is an impulse:

$$|u(t)| \le u^0, \qquad v(t) = V\delta(t - \tau), \qquad |V| \le v^0, \qquad \theta \in [0, 1]$$

Here V and θ are unknown: they are chosen by the second player (v) who maximizes J.

4. Both control and disturbance are impulses:

$$u(t) = U\delta(t - \tau), \qquad v(t) = V\delta(t - \theta), \qquad |U| \le u^0, \qquad |V| \le v^0, \qquad \tau, \theta \in [0, 1]$$

The first player chooses U and τ, whereas the second player chooses V and θ.

5. The control is subjected to the integral constraint, whereas the disturbance is bounded:

$$\int_0^1 |u(t)|dt \le u^0, \qquad |v(t)| \le v^0$$

Both control and disturbance are subjected to integral constraints:

$$\int_0^1 |u(t)|dt \le u^0, \qquad \int_0^1 |v(t)|dt \le v^0$$

Some results for all six cases obtained by the optimal control and differential games theories [2,3] are briefly summarized below. Here, indices $1-6$ correspond to the respective case and $k = u^0/v^0$:

$$\begin{aligned}
u_1 = -kv_1, \quad J_1 = (1-k)/2 \quad &\text{if} \quad 0 \le k \le 1 \\
u_1 = -v_1, \quad J_1 = 0 \quad &\text{if} \quad k > 1 \\
\tau_2 = 1 + k - (1+k^2)^{1/2}, \quad &J_2 = 0.5 + k^2 - (1+k^2)^{1/2} \\
\theta_3 = 0, \quad J_3 = 1 - 0.5k \quad &\text{if} \quad 0 \le k \le 1 \\
\theta_3 = 1 - k^{-1}, \quad J_3 = 0.5k^{-1} \quad &\text{if} \quad k \ge 1 \\
\tau_4 = \theta_4 = 0, \quad J_4 = 1 - k \quad &\text{if} \quad 0 \le k \le 1, \quad J_4 = 0 \quad \text{if} \quad k \ge 1 \\
J_5 = 0.5(1-k)^2 \quad &\text{if} \quad k \le 1, \quad J_5 = 0 \quad \text{if} \quad k > 1
\end{aligned}$$

The minimal values $J_i(k)$ for cases $1-5$ are shown in Fig.1. Case 6 occurs to be identical to case 4: here both optimal control and disturbance are impulses given at $t = 0$, and $J_6 = J_4$.

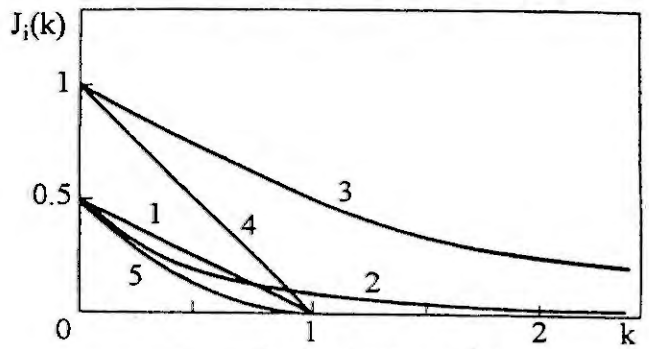


Fig. 1. Optimal values $J_i(k)$.

Note that, for the bounded disturbance, the impulse control gives better results than the bounded one, if $k < 0.75$ (here $J_1 > J_2$); if $k > 0.75$, the bounded control is preferable: $J_1 < J_2$. The multi-impulse control which corresponds to case 5 produces even better results for all k. For the impulse disturbance, the impulse control gives better results than the bounded one for all k: $J_3 > J_4$.

Time optimal control in the presence of disturbance

Consider the problem of time-optimal control for system (1) under the constraints (2). The initial state is arbitrary, and the terminal state is zero: $x(0) = x^0, \dot{x}(0) = \dot{x}^0, x(T) = \dot{x}(T) = 0$. The time-optimal feedback control minimizing the maximum $\max_v T$ (over all admissible $v(t)$) is given by

$$\begin{aligned}
u_\rho(x,\dot{x}) &= u^0 \operatorname{sign} \psi_\rho(x,\dot{x}) \quad \text{if} \quad \psi_\rho \ne 0 \\
u_\rho(x,\dot{x}) &= u^0 \operatorname{sign} x = -u_0 \operatorname{sign} \dot{x} \quad \text{if} \quad \psi_\rho = 0 \\
\psi_\rho(x,\dot{x}) &= -x - \dot{x}|\dot{x}|[2(1-\rho)]^{-1}, \quad \rho = k^{-1} = v^0/u^0 < 1
\end{aligned} \tag{3}$$

This control (3) is easily obtained by the approach of the theory of differential games [1,2]. The zero terminal state $x = \dot{x} = 0$ is always reachable, if and only if $\rho < 1$.

Let us compare this result with the simpler approach often used in practice. Suppose the disturbance is absent: $v = 0$ in (1); then the time-optimal feedback control is given by (3) with $\rho = 0$. We substitute this control into (1) and analyze the dynamics of the system

$$\ddot{x} = u_0(x,\dot{x}) + v, \qquad |v(t)| \le v^0 \tag{4}$$

in the presence of the bounded disturbance. It was shown [1] that the behavior of system (4) depends on the parameter ρ in the following way. Denote by ρ^* the golden section: $\rho^* = 5^{1/2} - 1 = 0.618...$ If $\rho < \rho^*$, all trajectories of system (4) reach the terminal state $x = \dot{x} = 0$ in finite time under any admissible disturbance $v(t)$. If $\rho = \rho^*$, there exist admissible disturbances such that the state $(x, \dot{x})$ stays in a bounded domain but never reaches the terminal point $x = \dot{x} = 0$. If $\rho > \rho^*$, then there exist admissible disturbances $v(t)$ such that the corresponding trajectories of system (4) are bounded and go to the infinity. These results are illustrated by Fig. 2. Hence, the simplified control ignoring the disturbances brings system (1) to the zero terminal state only if $\rho < \rho^*$, whereas the game approach is more efficient and achieves the same result if $\rho < 1$.

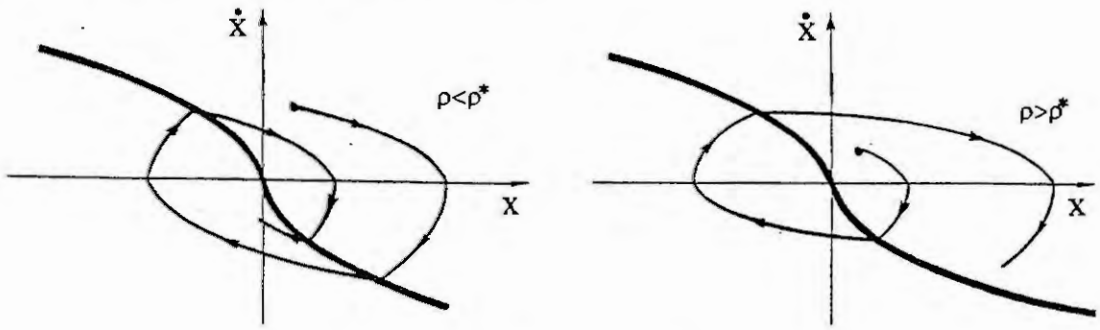


Fig. 2. Phase trajectories.

Time optimal control under complex constraints

Consider the time-optimal control problem for system (1) in the absence of disturbances ($v = 0$) and under more complicated restrictions imposed on control. Let the normalized equations of motion and constraints be

$$\begin{gathered} \dot{x} = y, \quad \dot{y} = z, \quad \dot{z} = u, \quad |z| \le 1 \\ u \le 1 \quad \text{if} \quad z \ge 0, \quad u \ge 1 \quad \text{if} \quad z \le 0 \\ x(0) = x_0, \quad y(0) = y_0, \quad z(0) = z_0, \quad x(T) = y(T) = 0 \end{gathered} \tag{5}$$

Here the magnitude of the control force z is bounded while its rate of change u is partly bounded. The force z may increase only gradually, at a finite rate, but it can be switched off instantaneously (Fig.(3)).

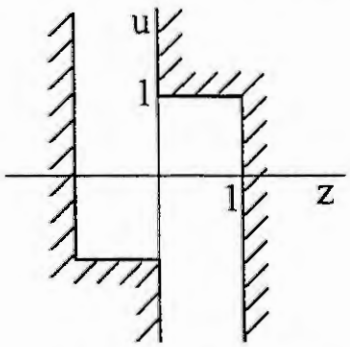


Fig. 3. Control restrictions.

This is not infrequently the case in practice, since deceleration is often implemented by means other than acceleration. The possible types of law governing the force $z(t)$ under these restrictions are shown in Fig. 4. It can be shown that, for any initial state in (5), the time-optimal control is given by one of laws $1 - 6$ from Fig.4 or by their mirror-image laws $z' = -z(t)$. Let the initial state x_0, y_0 lies in the domain D lying to the left of and below the curves Γ and Γ' shown by the thick curves in Fig. 5. Then this state belongs to one of six domains D_i shown by integers $i = 1, ..., 6$ in Fig. 5, and the respective control law corresponds to the same i in Fig. 4. Domains 1 and 2 are parts of the curve Γ below and

above the point $A = (-1/3, 1/2)$ in Fig.5, respectively. As $t \to T$, phase trajectories approach the origin touching the curve Γ_0 as shown in Fig. 4 by broken lines. For initial states to the right of and above the curves Γ and Γ' in Fig. 5, the control laws are mirror-images of laws $1 - 6$ from Fig. 4.

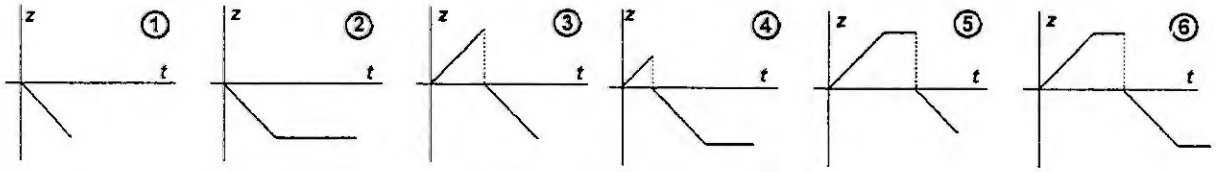


Fig. 4. Control laws ($i = 1, ..., 6$).

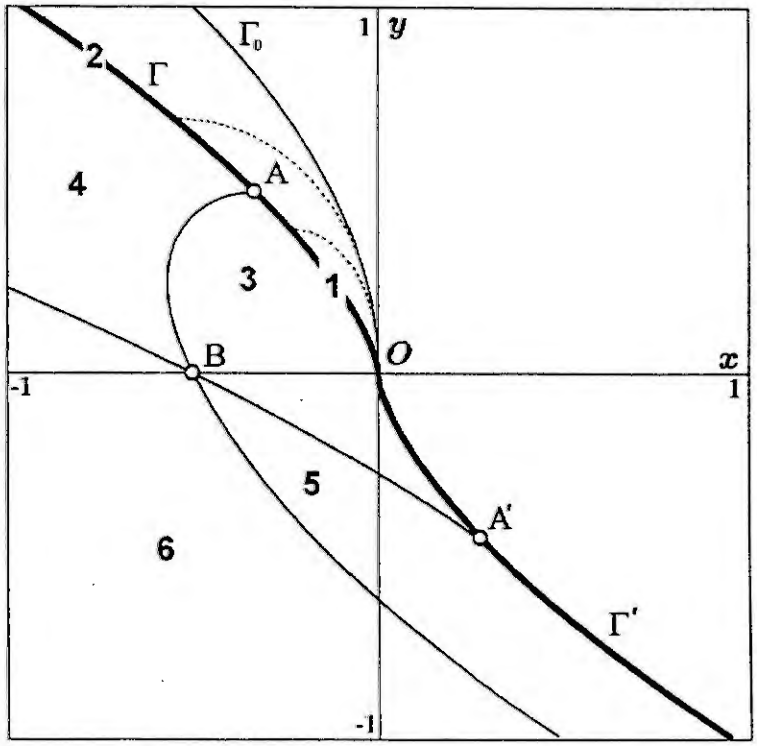


Fig. 5. Domains $D_i, i = 1, ..., 6$.

Conclusions

The results presented above show that, even for the simplest system (1), optimal controls can be rather complicated and depend significantly on the assumptions made. The optimal control laws are obtained in an explicit form and can be applied to nonlinear dynamical systems with many degrees of freedom in the frames of decomposition approach which reduces these systems to sets of subsystems with one degree of freedom similar to (1), see [1].

Acknowledgments. The work was supported by Russian Foundation of Basic Research (Grant 99-01-00258) and by INTAS-RFBR Grant 97-10782.

References

1. Chernousko, F.L., Decomposition and suboptimal control in dynamic systems. Optimal Control Applications and Methods, 14 (1993), 125-143.

2. Krasovskii, N.N., Game Problems of the Encounter of Motions. Nauka, Moscow, 1970.

3. Pontryagin, L.S., Boltyanskii, V.G., Gamkrelidze, R.V., and Mishchenko, E.F., Mathematical Theory of Optimal Processes. Wiley, New York, 1972.

Formal Specification of Dataflow Languages With Graph Grammars

M. Münch
Lehrstuhl für Informatik III, RWTH Aachen
52074 Aachen, Germany
muench@i3.informatik.rwth-aachen.de

Abstract. In this paper we present a graph grammar based approach for specifying visual dataflow languages. Graph rewriting is a useful tool for precisely specifying the syntax and semantics of a visual language. We developed a formal specification method with the help of the graph rewriting system PROGRES. We introduce a simple typed visual language HOTVLa for demonstrating the use of PROGRES and how to specify a model for a visual language.

1 Introduction

Visual languages are of increasing popularity and importance for many engineers. These languages can not only be found at university laboratories but also in industry. The most successful visual languages are based on the dataflow paradigm, i.e. data is passed from function (object, actor) to function via data connections (dataflows) (see [3] and [12] for examples). Another approach of visual languages is made up by grid-based icon rewriting systems. These systems are commonly used to program the behaviour of robots. However, the syntax and semantics of both kinds of languages are only informally defined.

Due to the graphical character of visual languages, an appropriate formalism to specify the syntax and semantics of those languages is the graph technology. Graphs play an important role in applied Computer Science. There are a lot of visual languages and environments which use graphs as their underlying data model (see e.g. [8], [9], [10]). Graph grammars and graph rewriting techniques as explained in the next section are a useful approach for our purposes of defining the syntax and semantics of a visual language. Graph grammars e.g. have been used for this definition in DiaGen [4]. However, our approach aims at a general specification of visual dataflow languages instead of considering limited application domains only as has been done till now.

In this paper we will concentrate on the modelling process of a visual language using a graph transformation system. We will show how to specify the alphabet and variables of a language grammar and the rules of that grammar graphically.

2 Graph Grammars

For the formal specification of a visual language we use graph grammars (see [6]). A graph grammar consists of the definition of the node- and edge-types of a family of graphs (atomic elements) and the transformations on these elements (rules). Graph nodes represent objects or concepts, graph edges represent relationships between them. For example, in a circuit diagram, graph nodes represent chips or gates or resistors, and graph edges are electrical connections. The rules of a graph grammar are actually graph transformations. In connection with the atomic elements they are used for creating or deriving sentences of the grammar. Furthermore, our graph model comprises some auxiliary information which is attached to nodes in form of attributes. With these three graph elements — nodes, edges, and attributes — we are able to represent complex, structured information at a convenient level of abstraction, as e.g. a diagrammatic representation of a visual language grammar.

Strictly spoken, a graph grammar is a set of productions that generates a language of terminal graphs and produces non-terminal graphs as intermediate results. However, we are interested in the specification of a visual language grammar and the generation of an application prototype serving as an editor, analyser and interpreter for this language. Therefore, we will use a graph transformation system instead of graph grammars only. A graph rewriting system is a set of rules that transforms one

instance of a given class of graphs into another instance of the same class of graphs without making the distinction between terminal and non-terminal results. Graph transformation systems are often used as visual graph manipulation tools while graph grammars are mainly used for synthesizing or recognizing graph-like data structures.

For specifying a visual language grammar we make use of the graph transformation system PROGRES[1] (see [7]) which is available as free software. The PROGRES system and language has already been used as the underlying fundament of a new approach to diagram parsing (see [5]), for defining the semantics of a visual database query language (see [1]), and many more projects. The PROGRES language offers several constructs for graph rewriting, querying, and building deterministic and non-deterministic control structures. The environment comprises a syntax-directed editor, a static analyser, and an interpreter for executing a specification. Furthermore, it is possible to compile a specification to C code to build a "stand alone" application with the help of a prototyping framework.

With these tools we are not only able to specify a visual language formally and describe the syntax and semantics by our specification but also to generate a "stand alone" editor and interpreter for this language.

3 Visual Language Specification

In this section we will give an example of how to specify a simple visual language with the graph rewriting system PROGRES. The language we have chosen is called HOTVLa[2]. It is a very simple language we have invented for demonstration purposes. HOTVLa comprises some simple mathematical functions, an if-then-else construct, and a function call construct. Fig. 1 shows a part of the language's alphabet definition, represented in the PROGRES system. This figure shows a part of the hierarchical language definition, similar to a graphical Backus-Naur-Form (BNF). All language elements are derived from HotVla_ELEM. HotVla_ELEM, CONTROL_STRUCTURE, and ARITHMETIC_OPS for instance are abstract nodes which are displayed as normal rectangles. Abstract nodes can be considered to be the non-terminal symbols (variables) of our grammar. The elements represented as rectangles with rounded corners are the terminal symbols (alphabet) of the visual language grammar, such as *Pair*, *Cmp*, ...

Every node can have attributes. For the sake of simplicity this is only shown for the *Pair* node. This node has three attributes: *fst*, *snd* and *TypeInfo*. The first two attributes are pointers to other HOTVLa elements which indicate the first and the second element of the tuple. The third attribute contains some type information for the *Pair* constructor which says that the function takes two parameters of arbitrary (and possibly different) type and returns a value whose type is the tupled combination of the input types. We have chosen a notation for this type information which is popular among functional programmers. This information is needed if we want to implement a type analyser for our HOTVLa language.

An equivalent piece of "Extended BNF" code would look like displayed in fig. 2. However, this notation cannot represent the attribute definitions in a similarly convenient way as our graphical representation.

After specifying the alphabet and variables by the schema of the graph grammar we define the rules of this language. Like in the well-known string grammars, our graphical rules consist of a left-hand side (LHS) and a right-hand side (RHS). The LHS specifies the graph pattern to be matched in the (visual) program definition (host graph) and the RHS specifies the pattern with which the previously matched pattern will be replaced. Fig. 3 shows an example of a rule to create a tuple out of two given values. Value1 and Value2 are given by the parameters to this rule (production) and matched by the LHS (nodes '1 and '2). In the next step this pattern will be extended by a new node 3' (Pair) and the two edges *fst*, and *snd*. The edges *fst* and *snd* correspond to the pointers as defined in fig. 1. The other two edges of type *data* represent the dataflows from the given HOTVLa elements to the tuple constructor. A simple string rewriting step could now update the type information as presented in fig. 1 by replacing the type information for "$*$" and "$**$" according to the type of nodes '1 and '2. Finally, we could derive the type of the whole function implemented in HOTVLa automatically. This will help us to define the semantics of the HOTVLa language.

Furthermore, this example also shows that our graph grammar is context-sensitive (type 1 grammars according to the Chomsky hierarchy)[3].

[1] PROgrammed Graph REwriting System

[2] Higher Order Typed Visual Language

[3] Of course, with PROGRES it is possible to define phrase structure grammars (type 0 grammars) so it is left to the user's discipline whether a grammar is context-sensitive or not.

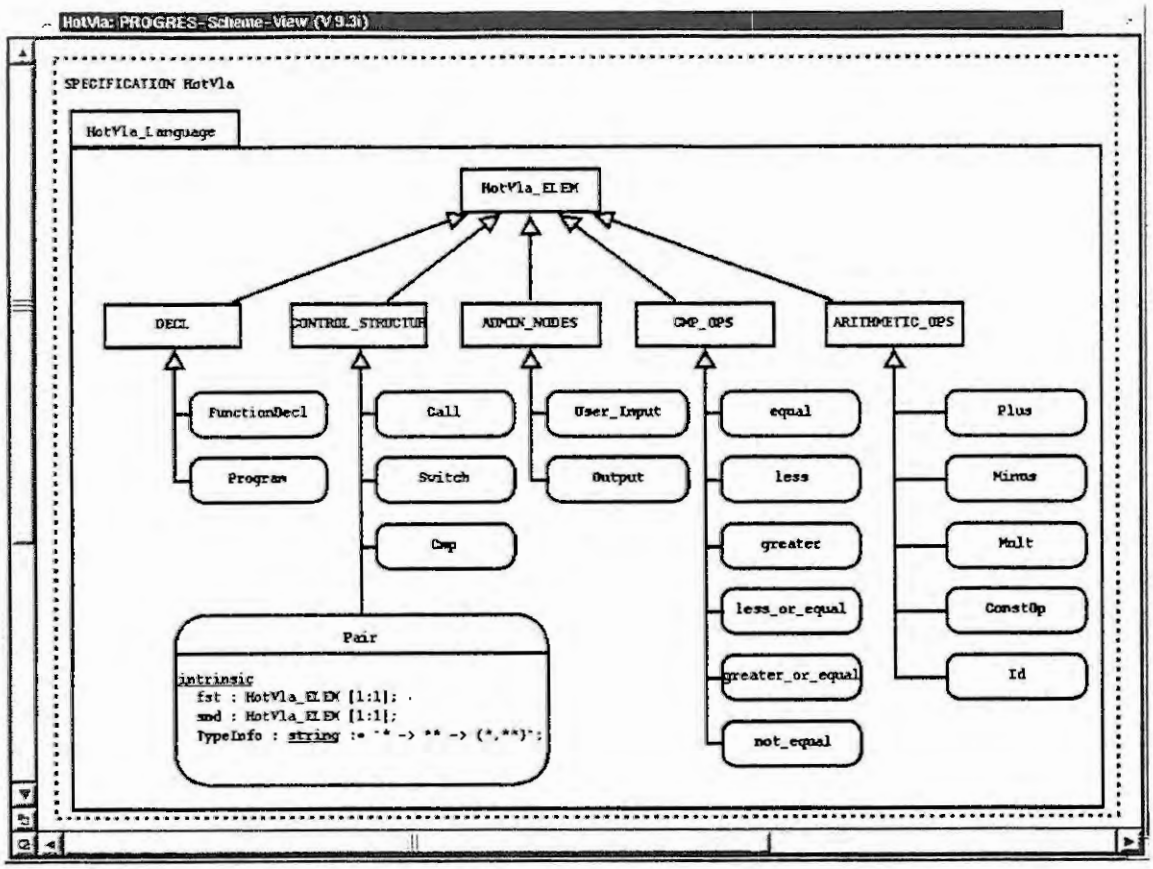


Figure 1: HOTVLa language definition

$$
\begin{aligned}
HotVla\_Elem &::= CONTROL\_STRUCTURES|ARITHMETIC\_OPS|\ldots && (1)\\
CONTROL\_STRUCTURES &::= Pair|Cmp|Switch|Call && (2)\\
ARITHMETIC\_OPS &::= ConstOp|Mult|Plus|Minus|\ldots && (3)
\end{aligned}
$$

Figure 2: HOTVLa EBNF language definition

Complex rules can be specified textually with the help of several control structures like deterministic and non-deterministic concatenations and disjunctions. With these constructs, accompanied by loops and conditional statements, we have also specified the behaviour of an editor or interpreter for our visual language. However, this does not belong to the specification of the visual language grammar but is a nice feature the PROGRES system offers. Due to the lack of space we cannot introduce the whole syntax of the HOTVLa language here.

For defining the semantics of our HOTVLa language we use the well-definedness of the PROGRES-language to deduce the semantics of the visual language expressions. This should be replaced by a proper approach in the near future.

4 Conclusion and Future Work

Graph grammars are an appropriate way of describing any visual dataflow language formally. We have shown briefly how to specify a simple visual dataflow language graphically. The schema of the graph grammar described the alphabet and the variables of our language and the operational part of the specification defined the rules. We have made use of the PROGRES system which also allows us to express complex rules that combine several atomic rules (such as *productions*) by control structures as e.g. loops, if-then-else constructs etc. The PROGRES system is able to generate a stand-alone

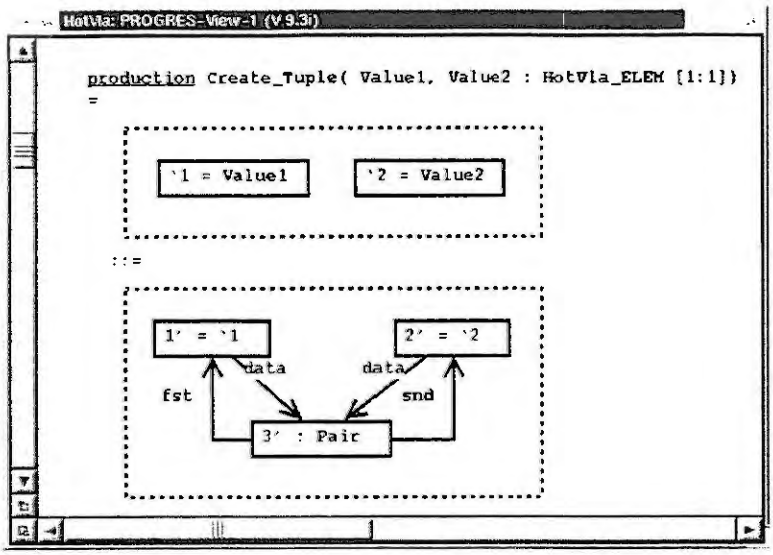


Figure 3: Grammar rule for creating a HOTVLa-tuple

application prototype out of such a specification. The defined rules can be exploited as editor functions in this prototype environment.

Furthermore, we have outlined briefly how to specify a typing analysis tool in the same approach. Meanwhile, we have also implemented an interpreter for our example language HOTVLa. Therefore, the resulting prototype of this specification is a fully functional editor, analyser, and interpreter for our visual dataflow language.

We will extend this research by examining several kinds of visual languages. The focus of this research will be put on component-based visual languages as they are widely used in e.g. process control engineering. Our main goal is to establish a library of PROGRES components (modules) for visual programming and modelling languages.

References

[1] M. Andries and G. Engels. Syntax and semantics of hybrid database languages. In H. J. Schneider and H. Ehrig, editors, *Graph Transformations in Computer Science*, volume 776 of *Lecture Notes in Computer Science*, pages 19–36, 1994.

[2] M. M. Burnett, A. Goldberg, and T. G. Lewis, editors. *Visual Object-Oriented Programming: Concepts and Environments*. Manning Publications Co., Greenwich, 1995.

[3] P. T. Cox, F. R. Giles, and T. Pietrzykowski. Prograph. In *[2]*, pages 45–66. 1995.

[4] M. Minas and G. Viehstaedt. Diagen: A generator for diagram editors providing direct manipulation and execution of diagrams. In *[11]*, pages 203–210, 1995.

[5] J. Rekers and A. Schürr. A graph grammar approach to graphical parsing. pages 195–202, 1995. Available from *ftp.wi.leidenuniv.nl*, file */pub/CS/TechnicalReports/1995/tr95-15.ps.gz*.

[6] G. Rozenberg, editor. *Handbook on Graph Grammars: Foundations*, volume 1. World Scientific, Singapore, 1997.

[7] A. Schürr. Introduction to PROGRES, an attribute graph grammar based specification language. In M. Nagl, editor, *Graph-Theoretic Concepts in Computer Science*, volume 411 of *Lecture Notes in Computer Science*, pages 151–165, 1990.

[8] *Proc. IEEE Workshop on Visual Languages (VL'89)*, Los Alamitos, CA, 1989. IEEE Computer Society Press.

[9] *Proc. IEEE Workshop on Visual Languages (VL'92)*, Los Alamitos, CA, 1992. IEEE Computer Society Press.

[10] *Proc. IEEE Symposium on Visual Languages (VL'93)*, Los Alamitos, CA, 1993. IEEE Computer Society Press.

[11] *Proc. IEEE Symposium on Visual Languages (VL'95)*, Los Alamitos, CA, 1995. IEEE Computer Society Press.

[12] G. M. Vose and G. Williams. LabVIEW: Laboratory virtual instrument engineering workbench. *BYTE*, 11(9):84–92, 1986.

Specification of Distributed Function Block Systems using UML

Ch. Diedrich, M. Riedl, Ch. Schmidt
Institut für Automation und Kommunikation Magdeburg
Steinfeldstr. 3, D-39179 Barleben

Abstract: The reduction of engineering cost is an up-to-date topic of automation systems. The function block paradigm is one of the key elements because it replace the application programming by application configuration. There are several function block models which are using different specification style. The specifications are for different components, e.g. for field devices or PLCs. At least all the components have to work together. The paper introduce the way how to describe FB models with the object oriented Unified Modeling Language (UML). Based on UML specifications of general parts of Foundation Fieldbus, PROFIBUS-PA and IEC 61499 a comparison is presented. Additionally it is shown how the UML FB applications can be simulated using the UML CASE tool Rational Rose and the SCADA tool iFIX of Intellution.

1 Introduction and application area

This paper presents investigation results for the instrumentation of industrial process measurement and control systems (IPMCS). Instrumentation means the transformation of a functional design into real devices which are connected with digital communication systems. This transformation includes a lot of different activities, for instance: Configuration of the communication system, allocation of the designed algorithms as Function Blocks (FB) into the devices, configuration of the data flow between the devices, configuration of modular devices and parameterization of the device applications to the specific process technology (commissioning of the devices, e.g. calibration, scaling). These activities are efforts additional to the classic algorithm design problems. Software tools are supporting the mentioned activities along with the life cycle of the devices in the system.

Function blocks seem to be a synonym for different entities of components and tools of IPMCS, i.e. both on-line operation and off-line engineering processes use the function block paradigm [1], [2], [3], [4], [5]. For on-line operation there are direct cooperation between function block instances, cooperation between function block instances and function block based proxies of remote functions and variables and cooperation between non-function block applications (e.g. visualization or maintenance) and function block instances. Off-line engineering is using most of all interface descriptions of encapsulated functions and variables which often also are called function blocks. The different areas are using different description methods for the function block specification, for instance function block diagrams, figures, tables or verbal text [6]. Therefore it is difficult to compare the different function block models. The strong consequence is, that manual written software adapter are necessary to cross the model borders between e.g. field device function blocks, IEC 61131-3 function blocks and objects in the control system. The need of additional resources and lower performance are the result.

Function Blocks are software entities which encapsulate functions. They are autonomous entities and it seems the function blocks are even objects. Therefore the object-oriented paradigm is a possible way to compare different function block models. Additional it is possible to use object-oriented analysis and design tools to specify function blocks and derive out of them software components for the operation system and the engineering tools.

The authors are using the object-oriented language and technology UML to specify the function block paradigm. This paper presents some results and is structured in the following parts:

- The mapping of the Function Block Type and its structure is done in section 2. The main focus of this step is the reference between the FB components and UML classes.
- The second step uses the found mapping principles to compare different function block standard specifications (section 3). The main focus of this step is to understand the differences between the models and to find missing items and problems.
- The third step is focusing on the use of the function blocks in applications (section 4). The main idea behind is to derive certain implementable representations to validate the FB specification. This is done in integrating the derived code in a simulation environment.

2 Function Block type specification as part of the device model

The device model contains the overall valid structure which combines the functional and hardware (i.e. the device as platform for the functions) aspects. The more hardware oriented components are the device and the modules (e.g. plug in I/O modules in a Remote I/O). The functional oriented components are variables, functions and the encapsulation of variables and functions, the blocks. There are many structural relations between these device components. A graphical representation is not precise enough to answer all questions of possible device configurations. Therefore a more formal specification is needed.

Note: The authors use the term block as generalisation of different kinds of function blocks. In the literature there is often no difference between the root of the function blocks types and their specialisation. This leads to misunderstandings.

The UML (Unified Modelling Language) class specification give the opportunity to specify components, their properties, their operation and their relation each other unambiguously. The graphical representation has to be mapped to UML language elements using the following approach:

- Components will become classes (Device is CDevice class, Function is CFuntion class, ...)
- The properties of the components becomes class attributes (e.g. Data Type of Variable)
- The relations between the components become relations between classes (e.g. functions are aggregated in blocks, ...)

The application related variable, function and block classes (CVariable, CBlock, CFunction) are integrated in the device structure consisting of CDevice, CArray and CModule. The functions (i.e. the automation algorithms) are aggregated to blocks but not aggregated in the CDevice class. That's why the algorithm implementations are hidden. That means that the functions are encapsulated in blocks which offer all or a subset of the input and output variables and parameters of the functions at their interface.

Devices may be hardware and software modular. Therefore the devices have the capability to contain a certain number of blocks (software modular) and modules (hardware modular). A device has something like a rack for the plug in the modules and the blocks. This is modeled with the CArray class. The CArray is part of the device and the blocks and modules can be plugged in the array. From a pure configuration point of view there is no difference between modules and blocks.

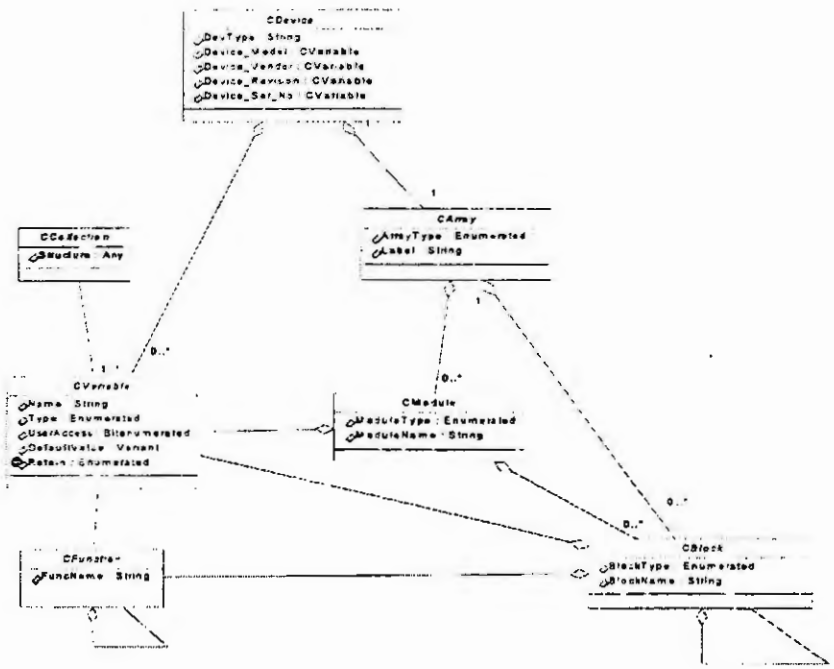


Figure 1. FB based Device Model as UML class diagram

At least a specific function block model specialize the classes CBlock, CFunction and CVariable. This is true not only for process related FBs, as for instance PID, but also for alarming, device configuration and FB scheduling.

A FB model is not fully described by class diagrams. Each class has a behavior, and the object of the class perform a data flow. Both can be modeled within the UML languages. Figure 2 shows an example of internal details of a Function Block.

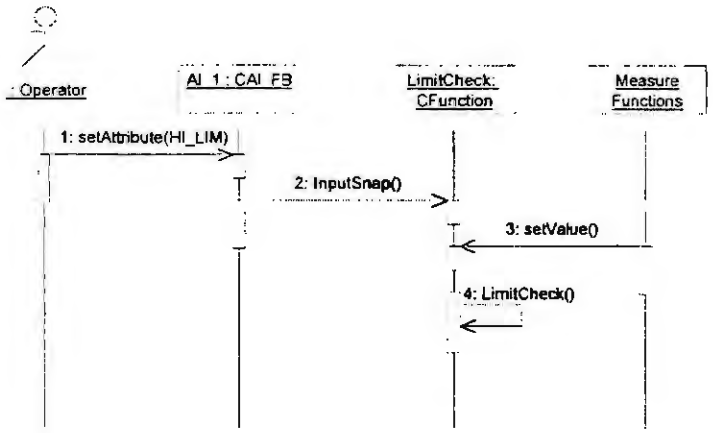


Figure 2. Example of a FB internal sequence

At least it is possible to specify the FB models in terms of class diagrams, sequence diagrams and state machines (not shown in this paper). The specification work using UML shows one fundamental difference between FBs models and the object-oriented paradigm among others. The data flow between objects are programmed in the application (calls of object operations), while the data flow between FBs are implemented by additional objects (so called link objects) which have to be configured during run time. The invocation of the FBs are organized in a so called FB Environment [7]. This environment organize implicit what have to be programmed in the object world. There is a strong reason for this approach, because process control applications have to be configurable during operation (e.g. Power plants).

3 Comparison of different FB models

Table 1 shows a comparison of FB models. It is based on the specification of general parts of the IEC 61499, Foundation Fieldbus Function Block Application and PROFIBUS-PA Profile standards in terms of UML. As already mentioned all specifications are mapped to the device model, which is introduced in section 2 of this paper. Then specialization is defined for the details of the standards.

Model element	Foundation Fieldbus	PROFIBUS-PA	IEC 61499
FB behaviour, i.e. functions	State machines, verbal descriptions	State machines, verbal descriptions	State machines for execution control of the functions, no other specifications
Input/Output data, i.e. variables	Mixed variable (Input and output), parameter (Input and output) and command (Input)	Mixed variable (Input and output), parameter (Input and output) and command (Input)	Differentiation between events (invocation of functions) and data (variables and parameters)
Block type	Unstructured FB types for a broad range of measurement, actuation and control functions	Unstructured FB types for a broad range of measurement and actuation functions	Unstructured (Basic FB) and hierarchical structured FBs (Composite FB), no functional specification
Connections between FB outputs and inputs	Separate objects, so called link objects	Separate objects, so called link objects	Connections are classes, but there is no statement about their behaviour
Control of functions*	Hidden in the detailed FB types	Hidden in the detailed FB types	Explicit modelled with the so called Execution Control Chart (ECC)
Control of FB*	Separate application entity, the so called System Management	Not yet	Interaction with a not specified Execution Function
Instanciation of FBs	Yes, in terms of Object	Not yet	Modelled as FB

types*	Dictionary modifications		application with a special Management FB
Declaration of FB types*	Yes, in terms of Object Dictionary modifications	Not yet	Modelled as FB application with a special Management FB

Table 1. Comparison between Function Block models

* - not described in the device model in section 2.

It is visible, that Foundation Fieldbus and PROFIBUS-PA FB applications are very similar. IEC 61499 is a framework to build detailed FB type specifications. Differences are most of all in the control and instanciation of functions and FBs. Differences in functional details are visible at the class diagrams, sequence diagrams and state machines. This paper discusses the model principles only.

4 Simulation of FB applications

One of the benefits using formal description technologies is to provide precise and unambiguously specifications for future implementations. UML CASE tools, like Rational Rose [8] offer the possibility to derive code out of the specification. In case of FB specification it is necessary to derive the code out of class diagram, sequence diagram and state machine to get an implementation framework which represents the main parts of the specification. The Rational Rose standard outputs are not covering the mentioned range of the models, but it offers a script language (Basic) to acquire all model elements. The Basic script is interpreted within the tool and generates the necessary code for further use. The generated code instanciates the FBs and their interaction in a simulation environment. The following steps are gone to simulate a FB application in a commercial tool:

1. Generating Basic code, which instanciates interacting objects according to the FB specification in UML.
   This is done using the Rational Rose script language (Basic). The result is a VB program.
2. Instanciating object in the simulation environment
   This is done using a VB tool, which interacts with the related interface of the simulation tool. As simulation tool the SCADA packet iFIX of Intellution [9] is used.
3. Simulation of the FB application
   The VB program instanciates variables, variable connections and visualisation instances in the SCADA package, i.e. a full application.

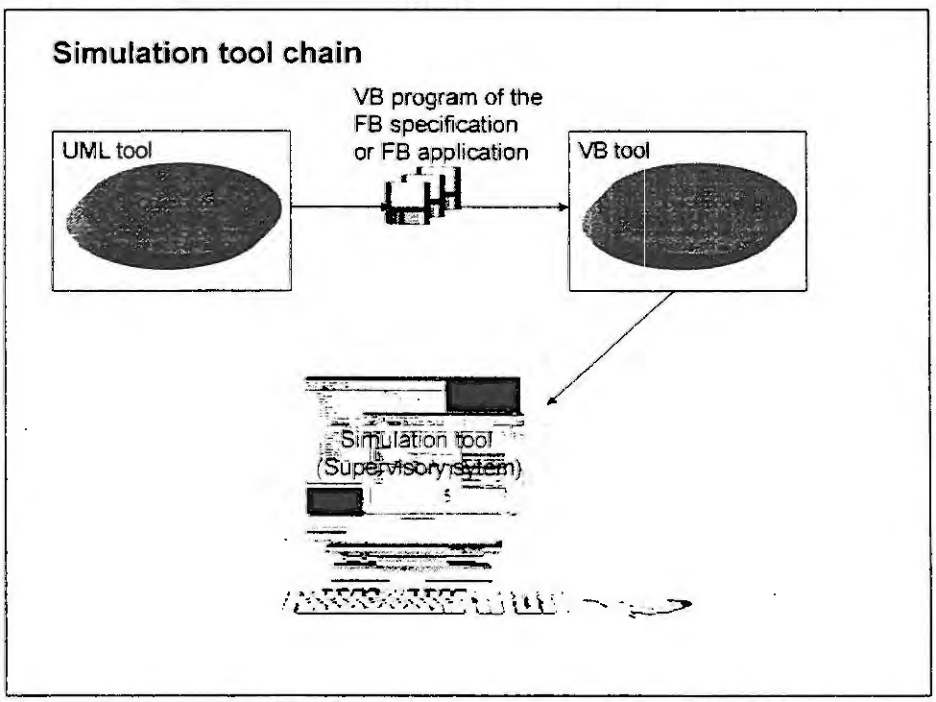


Figure 3. Simulation of FB application specified in UML

Using this approach it is possible to simulate UML based FB specifications. This is a very essential support for standardization activities or for FB specifications of device or system manufacturer.

5 Summary

The paper shows the way from informal FB specifications to UML based ones. This is a offer for those, who have to specify FB standards or are responsible for complex FB libraries. Based on UML descriptions of part of Fieldbus Foundation, PROFIBUS-PA and IEC 61499 there is a comparison of basic principles of these FB models. UML CASE tools offers the opportunity to derive code out of UML specifications. The authors offers a way how these FB specification can be instanciated in a simulation environment. For this approach the Rational Rose UML tool and the SCADA package iFix of Intellution is used.

6 References

[1] Polke, M.: Prozeßleittechnik, 2. Auflage, Oldenbourgh-Verlag, München 1994.

[2] IEC 61131 IEC 1131: Programming Language for Programmable Controllers, Geneva 1992. http://www.ab.cle.com/stds/iec/tc65bwg7/ .

[3] IEC 61499: Function Blocks for Industrial Process Measurement and Control Systems, Part 1 and 2, Committee draft Geneva 1998. http://www.ab.cle.com/stds/iec/tc65wg6/ .

[4] Foundation™ Specification, Device Description Language Specification, Fieldbus Foundation, Austin Texas 1996.

[5] PROFIBUS-PA Profile for Process Control Devices, Revision 3.0. PNO Karlsruhe 1999.

[6] Otto, P. and others: Fieldbus Profile Harmonization - Approach of NOAH EP 26951, FET'99, conference proceedings p. 423-428, ISBN 3-211-83394-3, Springer Verlag Wien New York1999.

[7] IEC 61804 Function Block for Process Control, Part 1 "General Requirements". Committee Draft, Geneva December 1997.

[8] Rational Rose: Unified Modeling Language, Version 1.1 http://www.rational .rose.com/uml, Rational Software Corporation, 18880 Homestead Rd, CA 95014, USA 1997.

[9] FIX-Software Intellution: http://www.intellution.com, Intellution Inc. Edgewater Drive, Norwood. MA 02062 USA.

MODELING OF SOFTWARE STRUCTURES IN PROCESS CONTROL SYSTEMS - AVOIDING BUGS BY USING GRAPH GRAMMARS

U. Enste and M. Kneissl
RWTH Aachen , Chair of Process Control Engineering
Turmstr. 46, 52064 Aachen, Germany
e-mail: udo@plt.rwth-aachen.de

Abstract. The aim of this research project is to support the engineering steps while designing function block types. The idea is, to extend the single-level class concept by introducing the concept of templates and design patterns into the function block technology. The described component based approach doesn't affect the process control systems in operation. To get a fomal model of the new features, the elements and the tranformation rules are described by a graph rewriting system. For the design of new function block templates or at least function block types, this formal language with well defined static semantics is helpful. Software for process control applications with poor error rates is expected.

Introduction.

The accepted basic concept to realize software applications in process control systems is to describe all necessary implementation specific functions with so called ‚function blocks'. A function block is a software unit, describing a method and its appropriated datastructure. Depending on the necessity of the data exchange between several function blocks, state variables of a block are distinguished between input data, output data and hidden state variables. Beside the principle of data capsulating, a single-level class concept is characteristic for the function block technology. The algorithm and the datastructure (syntax and semantic) are fixed in a 'function block type'. An application engineer can use these function block types to build a net of function block instances, which have their own data sets and a reference to their appropiated function block type. Due to these characterictic features, we indicate the function block technology as an object-based software concept.

Idea and Motivation

The idea of this research project is to extend the described object-based concept and to support the engineering steps in order to design function block types. Therefore, component based design patterns for building function block types are introduced. All modeling aspects of the new software architecture and all features to handle the new elements are described formally by graph-grammars. This allows us, to check the consistency of the developed architecture and to build prototypes checking the handling and benefits of the arised software system.

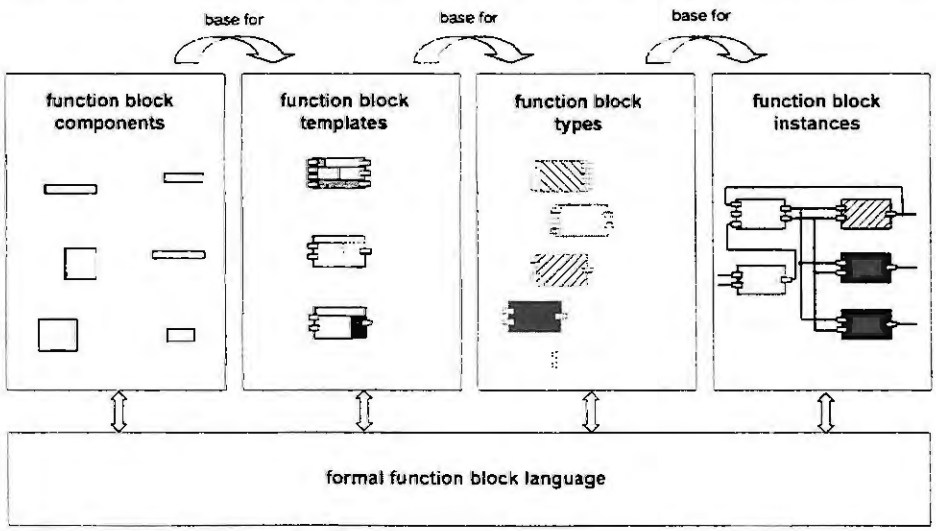


Fig. 1: Concept and its formal description to realize process control applications

The motivation for this research activity is based on two cognitions. First the control engineer must follow either normative or company-specific rules to design function block types. These rules are defined in textual or semi-formal specifications. Up to now there are no further modeling elements available for the control engineer to do the transformation from the textual specification to software code in an efficient and analyzable manner. The second cognition to motivate this research activity is the result of an analysis of several higher sophisticated function block types in industrial process control systems. The analysis showed, that a lot of functionalities inside a specific class of function blocks (blocks used for process control, blocks used for simulation,...) are encapsulatable and re-usable in a generic way.

Component based design patterns for building function block types shall evolve the engineering activities to design process control applications. Using components, which can be standardized (black boxes), generic or type-specific (white-boxes) templates can be developed, which pretend a specific structure and unified functionalities inside the function blocks of a specific function block class (see fig. 1). The idea is based on the principle of 'separation of concerns' [1].

Formal description using graph grammars.

A language with well defined static semantics has been developed to model sets of function block types with common properties. This language (‚FBComposed') is defined by the means of the graph rewriting system PROGRES (PROgrammed GRAph REwriting System)[1]. Graph grammars have been introduced into computer science in the early 70s and describe graph languages in a way similar to the way textual languages are formalized by string grammars. Using graph grammar techniques for the definition of the function block language makes the definition of FBComposed formal. Nevertheless, there is still an intuitive way to the specification of function blocks without deep knowledge of language theory.

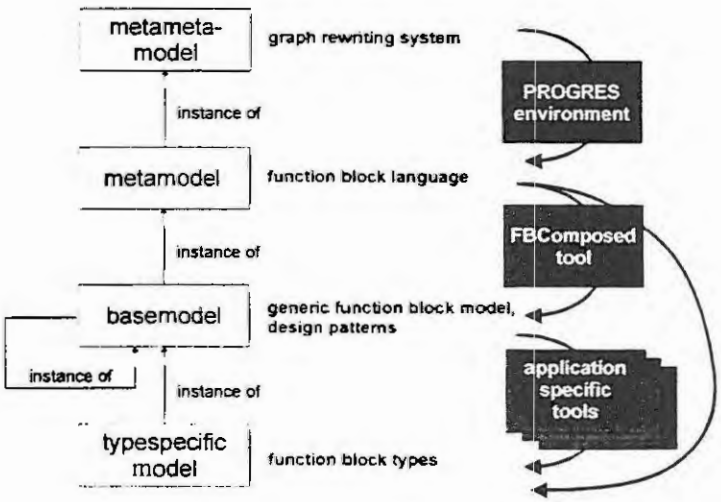


Fig. 2: Modeling levels and dedicated tools

The language FBComposed describes sets of function block models composed of smaller parts called "components". Hierarchical composition of components is possible. Each component has a defined boundary called "capsule" which encapsulates the component's implementation from its environment. Thus, the capsule can be considered as a black box view of the component. Components communicate by the means of signals which are exchanged along explicitly modeled connections between ports. Ports belong to the capsule of a component and mediate access to the component. No direct manipulation of a component's state is possible except by communication via connections across ports.

The implementation of a component is described by composing and connecting capsules as black box views of subcomponents. The results of this intermediate component construction step are called templates. A template can be instantiated forming a component by substituting a component for each of its capsules. This mechanism can be explained by comparing the capsule with a socket where a component can be plugged in. A template is

[1] developed at the department of Computer Science III at the Aachen University of Technology (see [7])

like a printed circuit board. Plugging components into all the sockets results to a new component which can be used thenceforth.

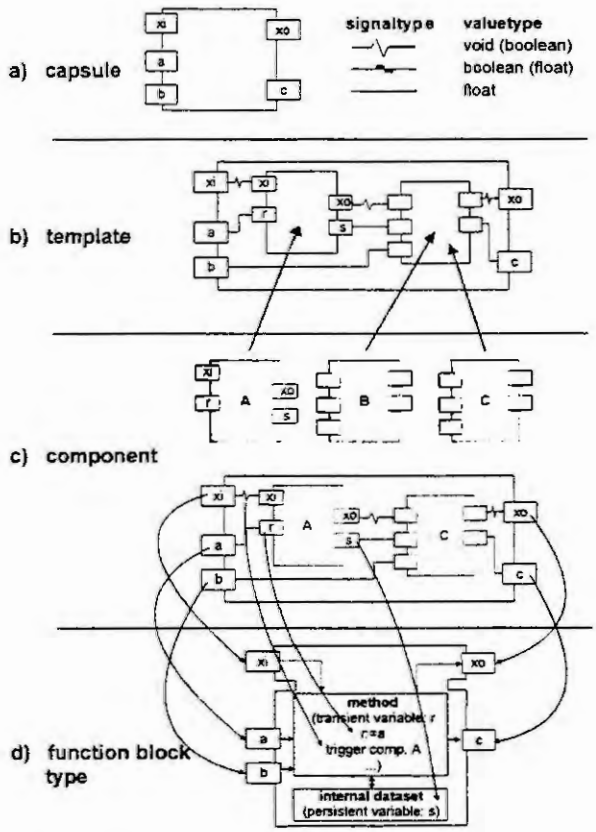


Fig. 3: Metamodel of the component based function blocks

The elements of the language "FBComposed" are mapped to nodes and edges of an attributed directed graph and vice versa. Each manipulation of the graph (and therefore each manipulation of the function block library document) is done by graph rewriting rules that are chosen to retain consistency in the framework. Additional constraints expressed by graph patterns including attribute conditions are used to point out forbidden and desired properties of the framework. These redundant constraints are also used to check the consistency of the metameta model itself. The development and modification of process control applications should be supported by a software tool that ensures compliance to the specification by following graph rewriting rules. A prototype of such a tool has been generated directly from the graph grammar specification (see fig. 2).

Template for batch-oriented process control units.

In process control, hierarchical control structures are used to design a network of sufficiently and asynchronously working software units. In such a process control model, superior control units send control instructions to inferior control units [4]. This kind of forward driven information exchange can be realized by standardized telegrams [2]. The handling of such control instructions, in particular the checking mechanisms to verify incoming instructions (syntactical and semantical checks) is a typical functionality which is worse to standardize by developing a generic component. This component considers also access rights of operators vs. automatic units. Beside this standardizable component the whole internal structure of batch-oriented control units can be described by the template technology for function blocks described above.

The structure with the functionality of each component can be outlined as follows (fig. 4 and [3]): The transaction control is the interface to the tasking of the function block system (like ECC in [5] but more generic). Activating this module means starting the algorithms of this function block. The transaction control represents a centralized control module inside a function block. It coordinates the control flow between all components inside the block. After checking an incoming instruction and verifiing its acceptance, the typespecific process control

logics can be activated. This is done based on a generic state machine which considers actual operating conditions which may prohibit the execution of the instruction. The information about the operating conditions can be fed in by typespecific signals. All these signals must be proved if they prohibit or enforce the starting or stopping of a specific process control logic. For this case, a typespecific component must be provided inside the template as a white box, where the typespecific signals can be mapped to the standardized signals of the generic component whith predetermined semantics. At least, one of several capsulated process control logics will be activated.

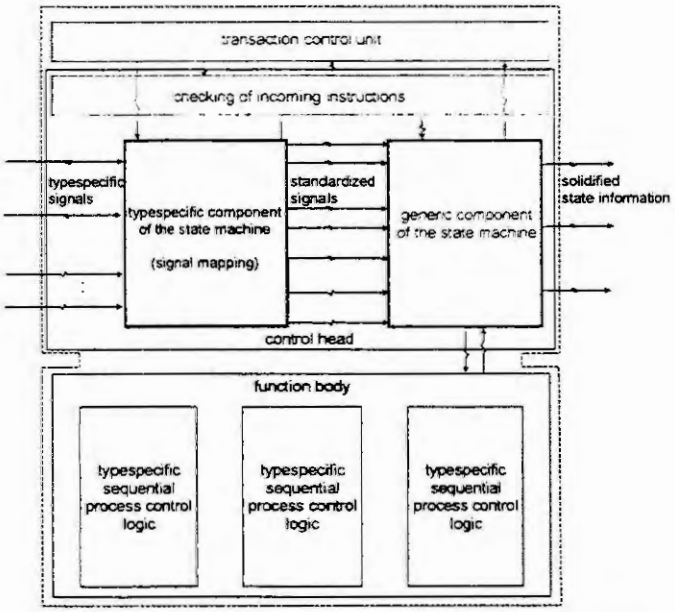


Fig. 4: Template for batch-oriented process control units

Conclusion

The presented approach extending the engineering-means for developing function blocks can be seen as ‚Programming in the Large' and leads to a software engineering environment and structure-oriented editors to develop generic function block models as well as function block types running in process control systems. Big efforts had to be done to apply a formal base to all conceptual ideas. The benefit of this formal base is a non-ambiguous specification and an avoidance of bugs which normally result on complex and error-prone software structures. The developed engineering tool ‚FBComposed' can be used by a process control engineer who has to develop function block templates or function block types as well as by normative committees who have to develop generic function block models (like [6]).

References

1. Czarnecki, K.: ‚Seperation of Concerns' – objektorientierte Frameworks und das generative Paradigma. OBJEKTspektrum 6, 1996.
2. Enste U., Fedai M.: Flexible process control structures in multi-product and redundant-routing-plants. 9th IFAC Symposium on Automation in Mining, Mineral and Metal Processing, Düsseldorf, 1998, 211-214.
3. Enste, U.; Epple, U.: Standardisierte Prozeßführungsbausteine - die Basis für Applikationsmodelle zur operativen Führung von verfahrenstechnischen Produktionsanlagen. VDI Bericht 1397, 1998, 505-512.
4. Epple, U.: Operational Control of Process Plants. In Polke, M.: Process Control Engineering. VCH-Verlagsgesellschaft. Weinheim 1994.
5. IEC TC65 WG6: Function Blocks for industrial-process measurement and control systems. Committee Draft IEC 61499-1, 1999.
6. IEC SC 65C WG7: Function Blocks for Process Control, Committe Draft IEC 61804-1, 1999.
7. Schürr, A.: Logic Based Structure Rewriting Systems. In: Ehrig, H.; Kreowski, H.-J. (Hrsg.): Lecture Notes in Computer Science 776, Springer, 1994, 341-357.

SYNTHESIS OF HIERARCHICAL PROCESS CONTROL SYSTEMS BASED ON SEQUENTIAL AGGREGATION

J. Raisch[1] and A. Itigin[2]
[1]Max-Planck-Institut für Dynamik komplexer technischer Systeme
Leipziger Str. 44, D-39120 Magdeburg, FR Germany
Tel.: (+49-391) 6117-523, Fax: (+49-391) 6117-501, email: raisch@mpi-magdeburg.mpg.de
[2]Institut für Systemdynamik und Regelungstechnik, Universität Stuttgart
Pfaffenwaldring 9, D-70550 Stuttgart, FR Germany
Tel.: (+49-711) 685-6296, email: itigin@isr.uni-stuttgart.de

Abstract. This contribution outlines a formal synthesis method for hierarchical control systems. It is based on a hierarchy of models describing the plant at various levels of abstraction and a decomposition of the overall specification. The proposed method captures several key requirements: it includes the notion of information aggregation between adjacent control levels; it allows for a combination of continuous and discrete event controllers on various levels of the hierarchy; complexity of the synthesis procedure and the resulting control scheme is considerably reduced when compared to an unstructured (and therefore non-hierarchic) approach; finally, it provides a mathematical guarantee that the specified hierarchical interaction between the different controller levels does indeed solve the overall problem. The method is illustrated by an example from process control, water level regulation in a two-tank laboratory experiment.

Keywords. Hierarchical control systems, hybrid systems, abstractions.

1. Introduction.

In many areas of application, process complexity has increased tremendously during the last few decades. This is mostly because process components are getting more tightly integrated to allow resources to be used more efficiently and to meet stricter environmental standards. Applying traditional, unstructured, control synthesis methods to such large-scale problems is certainly not advisable and often plain impossible: the complexity of the *synthesis procedure* usually reflects the problem complexity and quickly surpasses today's computer capabilities; problem complexity also translates into a complex *control structure*, which – if it were implementable – would be extremely hard to interprete.

Hierarchical control is an attempt to handle complex problems by decomposing them into smaller subproblems and reassembling their solutions in a hierarchical structure. Not surprisingly, it has been a popular topic within both academia and industry. In practice, heuristic approaches have been preferred. While they usually succeed in "breaking" the control task into problems of feasible dimension, they cannot guarantee that the overall solution does indeed meet the specifications. Formal approaches, on the other hand, have mostly been restricted to a small class of problems; typical assumptions are linear time-invariant plant models and quadratic cost functions (see e.g. [9]). There has also been a lot of recent activity in hierarchical discrete-event systems (e.g. [12]). Other approaches for less restricted problem classes have been reported in [2, 1].

In the following, we will describe an attempt to combine the advantages of both formal and heuristic approaches in a rigorous theory for the synthesis of hierarchical control systems. It is based on a hierarchy of models describing the plant at various levels of abstraction and a decomposition of the overall specification. It captures intuitive concepts like information aggregation between different levels of control, it allows the combination of continuous and discrete-event controllers within the same hierarchical structure, and it provides a mathematical guarantee that the resulting hierarchical structure does indeed solve the overall problem.

To embed continuous and discrete-event aspects within the same synthesis procedure, we need a broad mathematical framework. This is provided by J. C. WILLEMS' behavioural systems theory. Control and abstraction from a behavioural point of view are summarized in Section 2. In Section 3, we describe how to formulate a hierarchical synthesis approach within the behavioural framework. In Section 4, we show how it can be applied to a simple water level regulation problem.

2. Control and abstraction in a behavioural context

In WILLEMS' behavioural framework (see for example [10, 11]), a dynamical system is defined to be a triple $(T, W, \mathcal{B})$, where T is the time axis and W denotes the external signal "space". Let $W^T := \{w \mid w : T \longrightarrow W\}$ represent the set of all functions mapping T into W or, in other words, the set of all signals evolving on the chosen time axis T in the signal "space" W. Then, the *behaviour* $\mathcal{B} \subseteq W^T$ is defined to be the subset of signals that the model deems possible. Hence, for any non-trivial model, the subset relation will be strict.

Of course, when performing actual calculations, a finite-dimensional representation of $(T, W, \mathcal{B})$ is needed. Behaviours are, however, an extremely intuitive way of "thinking" about systems and their interaction. This is illustrated by a standard feedback configuration: consider a system ("plant model") with input $u(t) \in U$, (measurable) output $y(t) \in Y$, $t \in T$, and behaviour $\mathcal{B}_p \subseteq (U \times Y)^T$. It is to be controlled by feeding back y to u via a second system ("the controller") with behaviour $\mathcal{B}_c$. Then, the closed loop behaviour is given by $\mathcal{B}_{pc} = \mathcal{B}_p \cap \mathcal{B}_c$ — only signal pairs (u, y) that are compatible with the dynamics of both plant model and controller "survive" closing the loop. In the simplest case, closed loop specifications can be formulated as a "legal" set $\mathcal{B}_{spec} \subset (U \times Y)^T$ of signal pairs. The control task is then to "enforce" $\emptyset \neq \mathcal{B}_{pc} \subseteq \mathcal{B}_{spec}$ by finding (and realizing) a suitable $\mathcal{B}_c$ (Fig. 1). Now, suppose that controller synthesis for a system $\Sigma_p = (T, U \times Y, \mathcal{B}_p)$ is inconvenient (because, for example, realizations of Σ_p are tricky to handle). Hence, we want to perform the synthesis step on the basis of an approximation, or abstraction, $\Sigma_a = (T, U \times Y, \mathcal{B}_a)$. Clearly, a *conditio sine qua non* for Σ_a is that $\mathcal{B}_a \supseteq \mathcal{B}_p$. If this condition were violated, Σ_p could respond to a given input signal with an unacceptable measurement signal which would not be predictable by the abstraction. Hence, this unacceptable phenomenon could not be suppressed by a control strategy based on Σ_a – the abstraction would be useless as far as control synthesis is concerned. As illustrated in Fig. 1, this "abstraction condition" implies

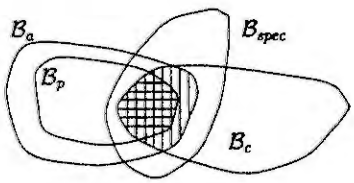


Fig. 1: Control and abstraction.

$$\mathcal{B}_a \cap \mathcal{B}_c \subseteq \mathcal{B}_{spec} \Longrightarrow \mathcal{B}_p \cap \mathcal{B}_c \subseteq \mathcal{B}_{spec}. \qquad (1)$$

One also needs to ensure that Σ_p and Σ_c are nonblocking, i.e. $\mathcal{B}_p \cap \mathcal{B}_c \neq \emptyset$. Assume this can be done (and in many scenarios this is straightforward or even trivial). Then, a controller which enforces the specifications for the abstraction Σ_a will also make the "base" model Σ_p obey the specifications.

3. Synthesis of hierarchical control systems

Functioning hierarchical control systems (in both technical and nontechnical areas) are often characterized by the following features: (i) the overall goal is decomposed into a high-level and a number of low-level specifications. (ii) The high-level specification is usually concerned with long-term developments, it involves aggregated signals, and it can be enforced on the basis of a coarse abstraction of the plant. (iii) Low-level specifications are usually concerned with short-term developments, they involve physical (measurement and control) signals, and enforcement of each low-level specification requires a detailed model of an appropriate plant component. We try to capture these features in a formal synthesis method. Fig. 2 illustrates that, within WILLEMS' behavioural framework, such a formalization can be interpreted as a generalization of a simple abstraction-based controller synthesis procedure: each broad "band" represents a set of output signals that, according to a coarse, abstract model, correspond to a given, possibly aggregated, input signal. Hence, on the basis of that model, one would be able to make the output avoid the forbidden area represented by the large grey boxes. A temporary tightening of specifications, represented by two additional black forbidden areas, can obviously not be enforced on the basis of that model, as none of the "output bands" fits through the small "legal gate". For that manoeuvre, one has to resort to a more detailed model that allows discriminating between "thinner threads" of output signal sets but needs only be defined in a narrow region.

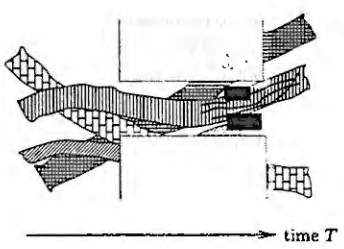


Fig. 2: Ilustration.

Retranslating this illustrative example into the formal domain, it becomes obvious that we have to perform two operations: *aggregration* and *subsystem selection.*

Let $\Sigma = (T, U \times Y, \mathcal{B})$ be a detailed model of the entire plant. On the *aggregation level*, there will only be condensed measurement and control signals and, often, a coarser notion of time: $\Sigma_{agg} = (T_a, U_a \times Y_a, \mathcal{B}_{agg})$, where $T_a \subseteq T$, and the aggregation of measurement and control signals is modelled by surjective functions $q_y : Y^T \longrightarrow Y_a^{T_a}$ and $q_u : U^T \longrightarrow U_a^{T_a}$. Hence, the aggregated model cannot distinguish lower-level measurement signals $\{y | q_y(y) = y_a, y_a \in Y_a^{T_a}\}$, and for each aggregated control signal u_a, any element from $\{u | q_u(u) = u_a\}$ can be selected on the lower level. Let $q(\mathcal{B}) := \{(q_u(u), q_y(y)) \mid (u, y) \in \mathcal{B}\}$. Then we require $\mathcal{B}_{agg} \supseteq q(\mathcal{B})$. Equality would imply that aggregation of control and measurement signals is the only loss of information when going from Σ to Σ_{agg}. If, on the other hand, $q(\mathcal{B})$ is a strict subset of $\mathcal{B}_{agg}$, additional modelling power has been sacrificed during the aggregation step.

Subsystem selection can be formalized in a similar way: suppose a particular subsystem model is only to be valid within a certain subset $Y_{si} \subset Y$. Introduce the "selection function"

$$\tilde{s}_i : Y^T \longrightarrow (Y_{si} \cup *)^T ; \quad \tilde{s}_i(y)(t) = \begin{cases} y(t) \text{ if } y(t) \in Y_{si}, \\ * \text{ if } y(t) \in Y \setminus Y_{si}, \end{cases}$$

where $*$ can be interpreted as an "out of range" symbol, and define $\mathcal{B}_{si} := s_i(\mathcal{B}) := \{(u, \tilde{s}_i(y)) \mid (u, y) \in \mathcal{B}\}$. Clearly, $s_i^{-1}(\mathcal{B}_{si}) \supseteq \mathcal{B}$.

The next step is to decompose the overall specification into a number of tasks that are to be solved on the aggregation and the subsystem levels. For this, we need to find $\mathcal{B}_{spec}^{agg} \subseteq (U_a \times Y_a)^{T_a}$ and $\mathcal{B}_{spec}^{i} \subseteq (U \times (Y_{si} \cup *))^T$, $i = 1, 2, \ldots N$, such that

$$q^{-1}\left(\mathcal{B}_{spec}^{agg}\right) \bigcap_{i=1}^{N} s_i^{-1}\left(\mathcal{B}_{spec}^{i}\right) \subseteq \mathcal{B}_{spec}.$$

Notice that this may tighten the specification, i.e. we may encounter a situation where, despite the original problem being solvable, no solution for the decomposed problem exists. This, however, is a price we expect to pay when trying to impose a hierarchical structure on a control system.

Now, the different parts of the overall control problem can be solved independently: we try to find a high-level controller $\Sigma_c^{agg} = (T_a, (U_a \times Y_a), \mathcal{B}_c^{agg})$ and subsystem controllers $\Sigma_c^i = (T, (U \times (Y_{si} \cup *), \mathcal{B}_c^i)$, $i = 1, \ldots, N$, such that $\mathcal{B}_c^{agg} \cap \mathcal{B}_{agg} \subseteq \mathcal{B}_{spec}^{agg}$ and $\mathcal{B}_c^i \cap \mathcal{B}_{si} \subseteq \mathcal{B}_{spec}^i$, $i = 1, \ldots N$. Clearly, the overall controller behaviour is

$$\mathcal{B}_c = q^{-1}\left(\mathcal{B}_c^{agg}\right) \bigcap_{i=1}^{N} s_i^{-1}\left(\mathcal{B}_c^{i}\right).$$

Therefore,

$$\begin{aligned} \mathcal{B}_c \cap \mathcal{B} &\subseteq \left(q^{-1}\left(\mathcal{B}_c^{agg}\right) \bigcap_{i=1}^{N} s_i^{-1}\left(\mathcal{B}_c^{i}\right)\right) \cap \left(q^{-1}\left(\mathcal{B}_{agg}\right) \bigcap_{i=1}^{N} s_i^{-1}\left(\mathcal{B}_{si}\right)\right) \\ &= q^{-1}\left(\mathcal{B}_c^{agg} \cap \mathcal{B}_{agg}\right) \bigcap_{i=1}^{N} s_i^{-1}\left(\mathcal{B}_c^{i} \cap \mathcal{B}_{si}\right) \\ &\subseteq q^{-1}\left(\mathcal{B}_{spec}^{agg}\right) \bigcap_{i=1}^{N} s_i^{-1}\left(\mathcal{B}_{spec}^{i}\right) \\ &\subseteq \mathcal{B}_{spec} . \end{aligned}$$

Hence, if blocking can be ruled out (i.e. if "base model" Σ, high-level controller Σ_c^{agg} and subsystem controllers Σ_c^i, $i = 1, \ldots N$, can "agree" on at least one common signal pair (u, y)), the overall control problem is solved.

For simplicity of presentation, we have restricted the discussion to the case of two control levels. It is quite obvious, however, that the procedure described above is transitive: once we know how to construct a 2-level hierarchy, we can also build an n-level hierarchy, $n > 2$. This will be demonstrated by applying our hierarchical synthesis concept to a simple two-tank-experiment. We emphasize that this example is *purely* for illustrational purposes. In particular, it so simple, that there is no real need for hierarchical control.

4. An illustrative example: the two-tank-experiment

The plant is shown in Fig. 3. It consists of two plexiglass tanks with identical cross sectional area $A = 154\text{cm}^2$ and height $L = 100\text{cm}$. The tanks are connected by a pipe; tank 2 is also equipped with an outlet pipe. Both pipes have cross sectional area $a = 0.5\text{cm}^2$. The pumps attached to tank 1 and 2 can be operated independently. In our set-up, they can only be switched between three discrete flow rates, hence $|U| = 3 \times 3 = 9$. Time is continuous ($T = \mathbb{R}^+$), but, as a further restriction on the control signal, u can only be changed at discrete instants of time, namely $t_i = i100\,\text{sec}$, $i = 0, 1, \ldots$. The output variables are the water levels in both tanks, hence $Y = [0, 100\text{cm}] \times [0, 100\text{cm}]$. The plant behaviour $\mathcal{B}$ is realized by two first order ODEs (nonlinear because of Toricelli's law and the obvious saturation for the water levels) under the above restriction for the input signal. The closed loop specification is as follows: we require $50\text{cm} < y_1(t_i) \leq 60\text{cm}$ and $30\text{cm} < y_2(t_i) \leq 40\text{cm}$ for $i \geq 5$, no matter what the initial water levels are. These specifications being rather weak, there seems to be no need for a detailed ODE-model. We will hence construct a discrete approximation Σ'_{app} or, equivalently, perform a pure aggregation step. It will turn out that the problem could be solved on the basis of Σ'_{app}, albeit at a high computational cost and giving rise to a fairly complex control structure. We will therefore take Σ'_{app} as the basis for a second aggregation step (which will result in an even coarser high-level model Σ_{app}) and a subsystem selection procedure (which will give rise to a subsystem model Σ_{si}). Both systems are much less complex than Σ'_{app}, and it will become clear that the overall control problem can also be solved within a hierarchical structure based on Σ_{app} and Σ_{si}.

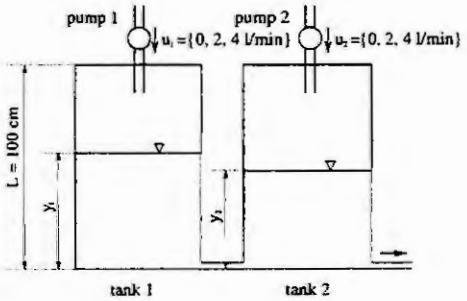


Fig. 3: Two-tank-experiment.

We now turn to the first step, which is pure aggregation: we construct a system $\Sigma'_{agg} = (T'_a, U'_a \times Y'_a, \mathcal{B}'_{agg})$, where $T'_a := \{t_0, t_1, \ldots\}$, $U'_a = U$, and Y'_a is a quantized version of Y. For both tanks, we assume ten quantization intervals, therefore $|Y'_a| = 10 \times 10 = 100$, and the aggregated measurement signal y'_a can take 100 symbolic values $y_a^{(1)}, \ldots y_a^{(100)}$ (see Fig. 4). Hence, $q'_u : U^T \longrightarrow {U'_a}^{T'_a}$ is a bijection (there a is one-to-one correspondence between input signals for Σ and Σ'_{agg}). Measurement signal aggregation is achieved by $q'_y : Y^T \longrightarrow {Y'_a}^{T'_a}$ where $q'_y(y)(t_i) = \text{quant}'(y(t_i))$ and the quantization function quant$'$ is illustrated in Fig. 4. q'_u and q'_y define q' and hence the aggregated behaviour $q'(\mathcal{B})$. We now use "strongest l-complete approximation" [3, 4, 6], an abstraction method originating in hybrid systems theory, to determine a suitable Σ'_{agg}. Application of this procedure results in the smallest behaviour $\mathcal{B}'_{agg}$ that (i) is a superset of $q'(\mathcal{B})$ and (ii) can be realized by a finite automaton whose state variable memorizes the last l values of the external signal (u'_a, y'_a). For $l = 2$, the state set of this automaton has 2171 elements, the number of transitions is 25236. We could now apply a formal (unstructured) synthesis procedure to check whether the (discrete) aggregation Σ'_{app} can be forced to obey the specifications and, if this is the case, to generate a suitable discrete controller. Based on RAMADGE's and WONHAM's supervisory control philosophy [7, 8], a synthesis procedure which is tailor-made for this situation has been described in [5, 3]. Because of $\mathcal{B}'_{agg} \supseteq q'(\mathcal{B})$ and the absence of blocking ([5, 4]), the resulting discrete controller will also "work properly" for the underlying system Σ.

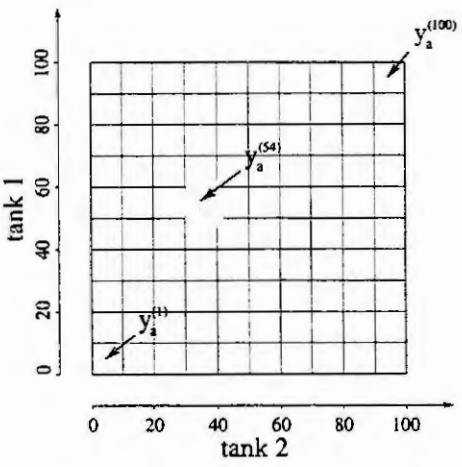


Fig. 4: Output quantization.

Instead of discussing this straightforward but complex solution, we opt to demonstrate our hierarchical approach by basing a second aggregation step and a subsystem selection operation on Σ'_{agg}. We first generate an aggregation $\Sigma_{agg} = (T_a, (U_a \times Y_a), \mathcal{B}_{agg})$ for Σ'_{agg}. We choose $T_a = T'_a$ and $U_a = U'_a$, i.e. Σ_a

is to operate on the same time axis and with the same input "space" as Σ'_{app}. The measurement signal is further condensed, however. Instead of 100 measurement symbols, which give a fairly accurate impression of the actual water levels, only a set of 9 symbols (and therefore very coarse measurement information) is available (Fig. 5). As in the previous step, aggregation functions for both control and measurement signal are defined in the obvious way: $q_u : {U'_a}^{T'_a} \longrightarrow {U_a}^{T_a}$ is the identity function; $q_y : {Y'_a}^{T'_a} \longrightarrow {Y_a}^{T_a}$ acts as $q_y(y'_a)(t_i) = \mathbf{quant}(y'_a(t_i))$ with the function quant : $Y'_a \longrightarrow Y_a$ partitioning Y'_a according to Figs. 4 and 5. No further loss of modelling power is desired, hence we choose $\mathcal{B}_{agg} = q(\mathcal{B}'_{agg})$, where q is uniquely defined by q_u and q_y. The resulting Σ_{app} can be realized by an automaton with 182 states and 2112 transitions.

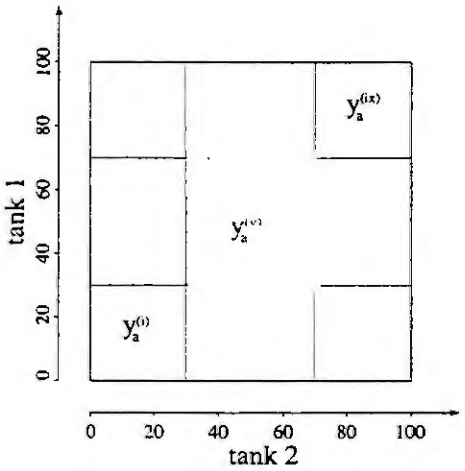


Fig. 5: Further aggregation.

Next, we describe how to determine a suitable subsystem $\Sigma_{s1} = (T'_a, (U'_a \times (Y_{s1} \cup *)), \mathcal{B}_{s1})$ for Σ'_{agg}. The motivation is to capture the modelling power of Σ'_{agg} in a limited area. The "interesting" output set Y_{s1} consists of all symbols from Y'_a that are mapped into $y_a^{(v)}$ under quant, i.e. $Y_{s1} := \{y_a^{(i)} | \mathbf{quant}(y_a^{(i)}) = y_a^{(v)}\}$ (Fig. 6). All other measurement symbols in Y'_a are lumped into the $*$-symbol. Definition of the selection functions $\bar{s}_1$ and s_1 is then completely analogous to the procedure in Section 3 (with T and Y replaced by T'_a and Y'_a, respectively), and $\mathcal{B}_{s1} := s_1(\mathcal{B}'_{agg})$. It turns out that the subsystem Σ_{s1} can be realized by an automaton with 285 states and 2112 transitions.

We still need to decompose the original specifications. Consider the following choice: (i) on the aggregated level, we require that $y_a(t_2) = y_a^{(v)}$. This completely defines the specification system $\Sigma^{agg}_{spec} = (T_a, (U_a \times Y_a), \mathcal{B}^{agg}_{spec})$; it can be easily realized by a simple finite state machine. (ii) On the subsystem level, we require that $y'_a(t_i) \in Y_{s1}$, $i = 3, 4, \ldots$, and $y'_a(t_i) = y_a^{(54)}$, $i = 5, 6, \ldots$; again, this completely defines a dynamical specification $\Sigma^1_{spec} = (T'_a, (U'_a \times Y'_a), \mathcal{B}^1_{spec})$, which is realizable as a finite automaton. Clearly, $q^{-1}(\mathcal{B}^{agg}_{spec}) \cap s_1^{-1}(\mathcal{B}^1_{spec}) \subset \mathcal{B}_{spec}$.

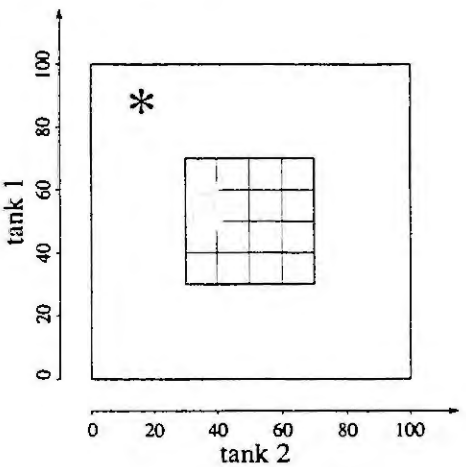


Fig. 6: Subsystem selection.

Hence, the only task for the yet to be synthesized high-level controller is to "drive" the water levels in both tanks into the area corresponding to the aggregated measurement symbol $y_a^{(v)}$ at time t_2, i.e. to ensure that 30cm $< y_1(t_2), y_2(t_2) \leq$ 70cm. Anything happening later in time is of no concern to the high-level controller. This is when the low-level controller has to come in. We want to synthesize it on the basis of the subsystem model Σ_{s1}, hence it needs to guarantee that the domain of Σ_{s1} will never be left: only measurement symbols from Y_{s1} are allowed to occur or, equivalently, the water levels must never escape the set [30cm , 70cm] × [30cm , 70cm]. After another two sampling intervals, the low-level controller must have "completed the job" by forcing the water levels into the set [50cm , 60cm] × [30cm , 40cm] and keeping it there for all future t_i, $i = 6, 7, \ldots$.

The only thing remaining to be done is synthesis of Σ_c^{agg} (on the basis of Σ_{agg}, Σ^{agg}_{spec}) and Σ_c^1 (on the basis of Σ_{s1}, Σ^1_{spec}). Again, this is a straightforward procedure using the method described in [5, 3]. Compared to an unstructured synthesis for Σ'_{app}, computational effort is reduced by about 95%. Moreover, complexity of the hierarchical controller is far lower than that of its unstructured counterpart. As blocking cannot occur in the considered framework ([5, 4]), we can guarantee that the resulting hierarchical control system will enforce the specifications for both Σ'_{app} (the detailed discrete abstraction) and Σ (the underlying continuous plant model). This is illustrated in Fig. 7, which shows a

simulation of Σ under discrete hierarchical control. It also indicates the time intervals where high-level control (based on the aggregation Σ_{app}) and low-level control (based on the detailed subsystem model Σ_{s1}) are active.

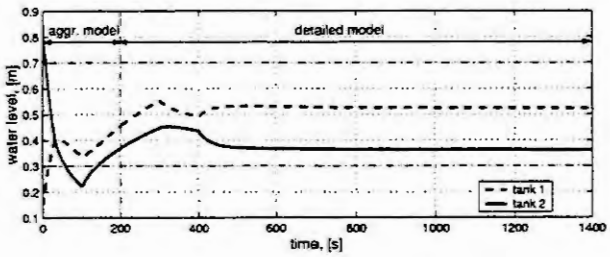


Fig. 7: Closed loop simulation.

5. Conclusions

In this paper, we have outlined a rigorous synthesis method for hierarchical control systems. It is set within WILLEMS' behavioural framework and captures some intuitive aspects commonly attributed to heuristic approaches only. For lack of space, discussion has been restricted to the conceptual level. Algorithms to realize certain aggregation steps have been reported in [4, 6]. Based on these algorithms, our hierarchical approach has been applied to a simple water level regulation problem.

Acknowledgements: The authors wish to thank *T. Moor* for many helpful discussions. Support from Deutsche Forschungsgemeinschaft through "Sonderforschungsbereich" SFB 412 is gratefully acknowledged.

References

[1] CAINES, P. E. and Y.-J. WEI: *Hierarchical hybrid control systems: a lattice theoretic formulation.* IEEE Trans. on Automatic Control, 43(4):501–508, 1998. Special Issue on Hybrid Systems.

[2] MESAROVIC, M. D., D. MACKO and Y. TAKAHARA: *Theory of Hierarchical, Multilevel, Systems.* Academic Press, New York, 1970.

[3] MOOR, T., J. RAISCH and S. D. O'YOUNG: *Supervisory Control of Hybrid Systems via l-Complete Approximations.* In GIUA, A., R. SMEDINGA and M. SPATHOPOULOS (Eds.): *Proc. WODES'98 - International Workshop on Discrete Event Systems*, Cagliari, Italy, 1998. IEE.

[4] MOOR, T. and J. RAISCH: *Supervisory Control of Hybrid Systems within a Behavioural Framework.* Systems and Control Letters, 38(3):157–166, 1999. Special Issue on Hybrid Control Systems.

[5] RAISCH, J. and S. D. O'YOUNG: *Discrete Approximation and Supervisory Control of Continuous Systems.* IEEE Trans. Automatic Control, 43(4):569–573, 1998. Special Issue on Hybrid Systems.

[6] RAISCH, J.: *A hierarchy of discrete abstractions for a hybrid plant.* APII-JESA, Journal européen des systèmes automatisés, 32(9–10), 1998. Special Issue on Hybrid Dynamical Systems.

[7] RAMADGE, P. J. and W. M. WONHAM: *Supervisory control of a class of discrete event systems*, SIAM J. Contr. Optimization, vol. 25, pp. 206–230 1987.

[8] RAMADGE, P. J. and W. M. WONHAM: *The Control of Discrete Event Systems.* Proc. of the IEEE, 77(1):81–98, January 1989.

[9] SINGH, M. G.: *Dynamical Hierarchical Control.* North-Holland, Amsterdam, 1980.

[10] WILLEMS, J. C.: *Models for Dynamics.* Dynamics Reported, 2:172–269, 1989.

[11] WILLEMS, J. C.: *Paradigms and Puzzles in the Theory of Dynamical Systems.* IEEE Transactions on Automatic Control, 36(3):259–294, 1991.

[12] WONG, K. C. and W. M. WONHAM: *Hierarchical control of discrete-event systems.* Discrete Event Dynamic Systems, 6(3):241–306, 1996.

DYNAMIC OBJECTS IN DISTRIBUTED CONTROL SYSTEMS

M. Fedai, U. Epple
Chair of Process Control Engineering, RWTH Aachen
Turmstraße 46, 52064 Aachen

Abstract. The article describes flexible structures for an operational control of a process plant. Therefore, a 'measure and resource model' is described as a formalism for control and administration of resources and actions. Measures are defined as a standardized interface between the production-, management-, and distribution control systems. The main idee of measures is, that each task in a plant system can be solved by several measure-objects. Measures are dynamic objects in distributed control systems and can get access or can give orders to several resources. Resources are intelligent and autonomous units, being able to solve well-defined tasks. For this kind of communication measures and resources have a standardized command interface to send and to receive short messages, called commands. By this measures and resources build up a hierarchical client server structure.

Introduction

In distributed control systems most software applications are realized by the software concept of a functionblock system. The functionblock system describes a modular and object oriented model where particular control functions can be implemented in modular functionblocks [5]. Functionblocks consist of an unambiguous name, input-, and output ports, internal variables and a method. The method describes the functionality of the realized control unit. The data exchange between functionblocks are realized with simple communication objects. Therefore, a communication object must be created and linked with the input port of a function block and an output port of another functionblock. Thus, a network of functionblocks can be built for controlling or simulating a process plant.

The described functionblock system with the rigid communication and static object structure does not fulfill the requirements for a dynamic control of a process plant. Function units are static and permanently present, whether they are used or not used. Some function units are only needed rarely. Static communication objects must be created and coupled with ports of functionblocks and function units can only communicate within a local FB System. The consideration of all events in an implementation of a functionblock makes it complex and delicate for errors.

Measures as dynamic objects

The measure and resource model describes a powerful concept for dynamic control of a process plant. It describes a flexible and technology invariant model, which is applicable for different tasks [6]. The model of dynamic objects represent a logical interface between the production-, management-, and process control level. The main idee of this concept is, that each task in a process plant can be solved by certain measure-objects, with the aid of plant resources. Different requirements, for example recipe based process control, logistic product management or maintenance can be realized by this model. It describes a formalism for controlling and administration of resources and measures.

Dynamic objects as here described in the measure and resource model are intelligent and temporary unit objects in the FB System with a standardized communication interface. Measures are dynamic objects in a distributed control system and represent higher control-level function units. They have type-specific operating instructions and a standardized communication interface and can perform their task by issuing orders to resources or other function units. Therefore measures may require and allocate various resources. They can be handled and started either by a plant operator or by an event or by an other higher level control unit. During the lifecycle of a measure it issues orders to function units, collects data, records protocols and reports actions. They are performed and supervised by a process control system. Dynamic objects can be started at any time by an event or by the plant operator. All dynamic objects are autonomous and independent function units.

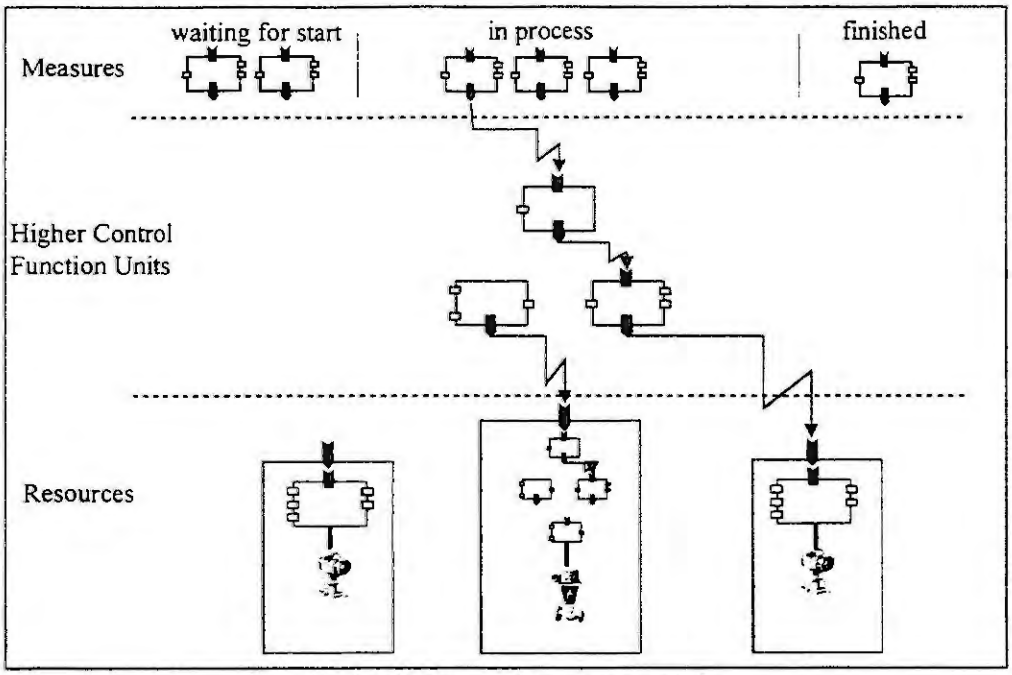


fig. 1: Measure and resource model

Resources are modular intelligent function units with certain characteristics. They consist of a plant element and an appropriate control unit. They represent the functional ability of a plant and are permanently present whether they are processing an order or not. They can be scheduled and planed independently of each other. Resources have also a standardized command interface to receive orders from higher control level function units. They can receive and process orders independently and can manage themselves. The planning and scheduling of resources are accomplished by higher-level function units.

Flexible communication structures between different type of objects

For a flexible communication between different type of objects and between different systems a standardized communication interface is needed. One example of a powerful communication is the command oriented communication. A command is defined as a short dynamic message, which can be sent at any time to objects with the command interface.

The idee of this communication concept is to build a hierarchical process control structure, as in a management, where higher level control units can send dynamic commands to inferior function units. These inferior units can process the command independently and can also send commands to other function units. This type of communication represents a typical client-server communication [4]. All Objects with the command interface have the following additional features:

- An open, standardized communication interface
- An allocation mechanism
- List of generic acceptable commands
- List of type-specific acceptable commands
- Internal method for examination of incoming commands

A significant feature of the command interface is the allocation mechanism. Each Object with the command interface, so called process control object, must be assigned to a higher level control unit, as in a management. This means process control objects only accepts commands from the current occupying control unit. Commands from all other function units are denied. The allocation and deallocation of objects can be broadcasted with the generic commands 'OCCUPY' and 'FREE'. By an allocation of a function unit the name of the higher control unit is registered in a process control unit. After the successful performance of a task, process control units must be deallocated by the registered user.
Only the plant operator is allowed to allocate a already allocated function unit. Because of the security of the plant, he must be in position to send a command at any time.
For an uniform receiving and sending of commands it is obvious to standardize the communication interface. Each command sent to an inferior function unit consists of three elements : Sender-ID, Command-type, and Command-value. The Sender-ID describes the name of the current sender, and the command-type with the

command-value are the real command. Process control function units have the following list of generic commands which they can accept:

- OCCUPY, registers the name of the current sender
- FREE, deletes the registered name of the sender
- TIOP, take in operation
- TOOP, take out of operation
- RESET, goes to the basic state

The plant operator can define his own type-specific commands, for example "ON, OFF, CONTROL; PUMP". The incoming commands are verified by process control function units by themselves, as in a client server system.

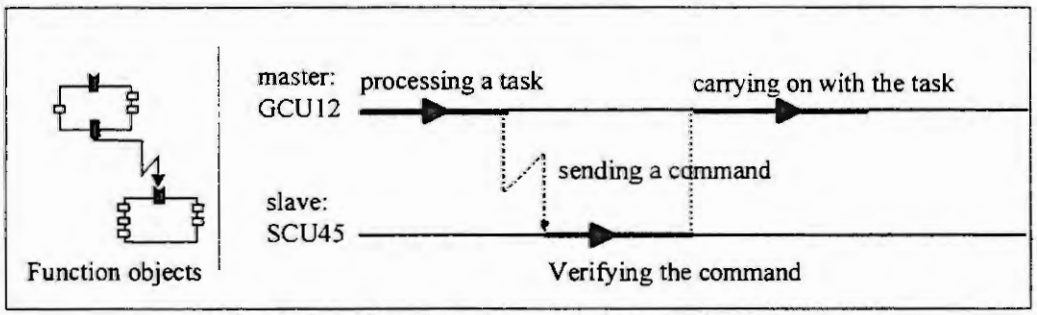


fig. 2: Broadcasting of commands in local systems

As shown in fig. 2 a higher-level group control unit (GCU12) sends a command to the single control unit (SCU45). The broadcasting is realized within the method of the GCU12 with the command COM( ). At first the system searches the function unit with the name SCU45. If the unit is found in the local system, the name of the target unit is exchanged with the own name, and sent to the unit. The verification-method of the single control unit (SCU45) checks the incoming command with the allocation and the generic and type-specific command list. If the command is accepted, it is copied into the command port and the group control unit (GCU12) can carry on with its task. Otherwise it is denied, and the command is ignored.

As dynamic objects has the need to send and receive messages or process values in distributed systems, a new client object is necessary to organize and to manage the communication. The client function unit represents the communication to a remote system. Each remote system must be represented by a client unit. Client objects create, and open a logical communication channel to the remote system and manage the asynchronous requests and responds. It collects all requests, composes them to one service package and sends them cyclically to the remote system. Thus, the communication frequency on the net is minimized.

The communication between different systems is realized with the open, platform and vendor independent communication system ACPLT/KS (Aachener Prozeßleittechnik/Kommunikationssystem) [1], [2]. It has generic services to read/write variables, e.g. process values, and to create/delete objects in distributed systems. The services can be processed asynchronously, i.e. function units can send a request and carry on with their task and receive later a respond. The data transfer is realized with the standardized TCP/IP protocol.

A distributed communication within the FB System can either be realized by enhanced communication objects, or dynamically broadcasting commands directly, within the function units, to the client unit. The enhanced communication objects send data to the client unit and receive them from the client. Therewith the existing methods of function units must not be changed. They can receive and send values with enhanced communication objects.

Based on this client-server communication structure, objects can now communicate with other objects in local and distributed systems. All function units can send messages and activate objects in other systems.
An object in a process control system can send a command to a maintenance measure in a distributed system and can start it. The measure unit can perform its task by issuing commands to resources. After the started measure unit is finished it is archived and deleted form the system.

Operating states of dynamic objects

Measures as dynamic objects are temporary function units. They are created by a plant operator and loaded into a scheduling system. The scheduling system organizes all needed resources and measures. Measures in the scheduling system can be loaded up into the performance system and wait until they are started or deleted by an operator or an automatic function unit. After a dynamic object is started it is also an object of an archive system. Up to now the measure cannot be deleted. All actions are reported and saved by the archive system. Measures can perform their task by issuing orders to resources or other function units. Therefore measures may require and allocate various resources. Complex plant tasks can be realized by several simple single measures. Measures can be, for example: "Pump x liters in boiler A.", "Test the safety devices in the plant." or "Produce x tons of the product Y in according to recipe Z.". During the lifecycle of a measure it issues orders to function units, collects data, records protocols and reports actions. After a measure is finished it is reported and archived by an archive system.

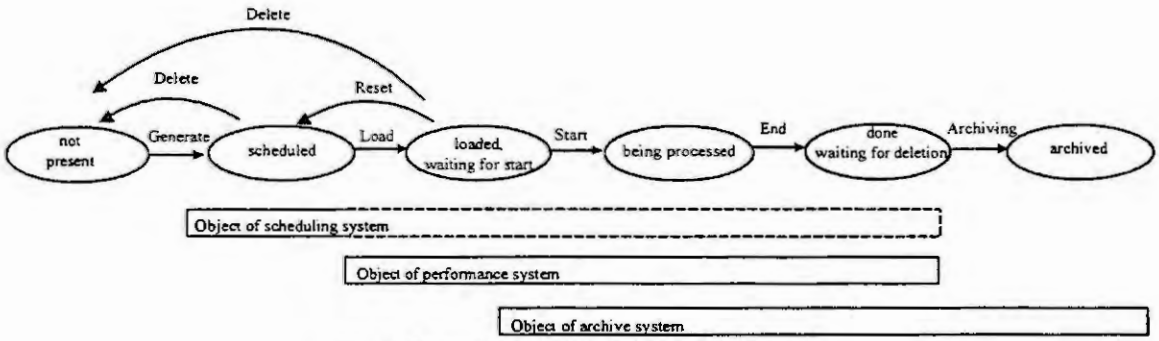


fig. 3: Operating states of dynamic objects

Measures may be used mainly for periodical and dynamic tasks. Cyclical maintenance tasks, for example a monthly check up of plant devices, can be performed automatically by measure units. They can be started by a higher level control unit or an event, and executed and supervised by a performance system. All completed and accessed measures can be viewed in an archive system. By this, a supervision of process plants or event oriented tasks can be performed with measure objects.

Summary

The flexible command oriented communication structure enables a flexible control of process plants. By this, a hierarchical process control structure can be built. Dynamic Objects, called measures, can perform event controlled, or cyclical tasks in a flexible way. Measures can also start and communicate with other measure objects in distributed systems. They can perform their tasks by issuing commands to allocated resources. Therefore, the measure and resource model with the command oriented communication represents a powerful and uniform concept for an operational control of process plants.

References

1. Albrecht, H., and Arnold, M., and Kneißl, M., Das PLT-Internet – Prozeßdaten unternehmensweit. In: Proc. Echtzeit '97/iNet'97, 210 – 215.
2. Arnold, and M., Epple, and U., Polke, M., Unternehmensweiter Zugriff auf Prozeßdateninformationen mit dem "PLT-Internet". Automatisierungstechnische Praxis 39 , (1997) 1.
3. Balzer, D. and Epple, U., Technologieinvariante Prozeßführungsmodelle. GMA Fachbericht 5, 1994.
4. Enste, U., and Fedai, M., Flexible Process Control Structures in multi-product and redundant-routing plants. In: Proc. 9th IFAC Symposium on Automation in Mining, Mineral, and Metal Processing, 1998.
5. Epple, U., Die Bausteintechnik – Grundlage einer objektorientierten Netzstruktur für die Prozeßleittechnik, In: VDE-Kongreß 20.-23.1.1993, Berlin.
6. Epple, U., Operational Control of Process Plants. In: Process Control Engineering, (Edited by Polke, M.) Weinheim, New York, Basel, Cambridge, Tokyo, 1994.

3-D MATEMATICAL MODELS FOR FINITE ELEMENT CALCULATIONS OF DYNAMICS AND STATICS OF MACHINERY

O. V. Repetski * and H. Springer
Technical University Vienna,
Wiedner Hauptstrasse 8-10, A-1040, Vienna

Abstract. Mathematical models based on the principles of the three-dimensional finite element method (3-D FEM) for static stresses, natural frequencies and forced vibrations of rotor structures are presented and verified. New integration methods for strength calculations of rotating turbomachinery components are worked out.

Theory.

In any finite element analysis a numerical solution must be carried out for a number of Gauss points to be specified for the integration of the stiffness and mass matrices. The integration order employed for the calculation significantly influence its accuracy [1,2,3].

The approach as used in "selective" integration is to split the stiffness matrix [K] into two separate componentss:

$$[K] = [K_A] + [K_B].$$

Reduced integration is used for one part of the stiffness matrix, and complete integration for the other one. Reduced integration is used for that part of the stiffness matrix which has the greatest influence on the accuracy of the element while complete integration of the other part supplies sufficient rank for the overall matrix to prevent any spurious mode. The difficulty is in deciding the best way to split [K]. Two approaches which retain this feature are:

1.Splitting [K] into a direct strain stiffness matrix and a shear strain stiffness matrix;

2. Separating [K] into a so-called "volumetric" stiffness matrix and a "distortional" stiffness matrix.

Using the first approach, the elasticity matrix for isotropic material can be written:

$$[D] = [D_{dir}] + [D_{sh}], \tag{1}$$

with

$$[D_{dir}] = \frac{E}{(1-\nu)(1-2\nu)}\begin{bmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ \nu & (1-\nu) & \nu & 0 & 0 & 0 \\ \nu & \nu & (1-\nu) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad [D_{sh}] = \frac{E}{(1-\nu)(1-2\nu)}\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & (1/2-\nu) & 0 & 0 \\ 0 & 0 & 0 & 0 & (1/2-\nu) & 0 \\ 0 & 0 & 0 & 0 & 0 & (1/2-\nu) \end{bmatrix}.$$

The total stiffness matrix $[K]$ can then be composed of a direct strain stiffness matrix $[K_{dir}]$ and a shear strain stiffness matrix $[K_{sh}]$ such that

$$[K] = [K_{dir}] + [K_{sh}] = \int_V [B]^T [D_{dir}][B]dv + \int_V [B]^T [D_{sh}][B]dv, \tag{2}$$

where $[B(x, y, z)]$ is a strain-displacement matrix.

When using the second approach, it should be considered that the total strain energy U is the sum of the volumetric strain energy and the distortional strain energy,

$$U = U_{vol} + U_{dist} \quad \text{and} \quad [D] = [D_{vol}] + [D_{dist}]. \tag{3}$$

* Currently, the author is a visiting Lise Meitner scholar from Irkutsk State Technical University at Technical University of Vienna

The stiffness matrix $[K]$ can then be composed of a "volumetric" stiffness matrix $[K_{vol}]$ and a "distortional" stiffness matrix $[K_{dist}]$ in the form

$$[K] = [K_{vol}] + [K_{dist}] = \int_V [B]^T [D_{vol}][B]dv + \int_V [B]^T [D_{dist}][B]dv, \tag{4}$$

with

$$[D_{vol}] = \frac{E}{3(1-2\nu)} \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$[D_{dist}] = \frac{E}{(1+\nu)} \begin{bmatrix} 2/3 & -1/3 & -1/3 & 0 & 0 & 0 \\ -1/3 & 2/3 & -1/3 & 0 & 0 & 0 \\ -1/3 & -1/3 & 2/3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/2 \end{bmatrix}.$$

The matrices K_{vol} and K_{dist}, K_{sh} and K_{dir} can be integrated separately by using different order of numerical integration for each matrix.

A new method of numerical integration called "mixed integration" was developed in [1,2]. This method involves carrying out the numerical integration of the stiffness matrix twice. The first time by using a reduced integration (2x2x2 rule) to form the matrix $[K_r]$, and the second time by using a complete integration (3x3x2 rule) to form the matrix $[K_c]$. The final stiffness matrix is then given in the form:

$$[K] = (1-\beta)[K_r] + \beta[K_c], \tag{5}$$

where β is a weighting factor (β=0.0001 to 0.01).

The idea behind the method is that a spurious mode shape produced by reduced integration could be predicted by complete integration according to the following features: 1) the natural frequency calculated for the spurious mode is usually higher than that calculated by reduced integration; 2) the spurious mode has a physically implausible nature. On the other hand, the natural frequency calculated by reduced calculation and by complete integration for a correct (non-spurious) mode would not differ by large amounts. Adding a small contribution from the complete integration stiffness matrix, $[K_c]$, to the reduced integration stiffness matrix, $[K_r]$, should have the effect of greatly increasing the frequencies of any spurious mode , virtually without any change in frequency of the correct mode. The improved accuracy of reduced integration for the correct mode should therefore be retained with mixed integration but the spurious modes have their frequencies pushed beyond the range of interest.

A new "combined" method of integration using the reduced integration for plate-shell parts and the complete integration for three-dimensional parts of the finite element model was proposed for testing the structures composed of two-dimensional (plates, shell) and three-dimensional parts. The 3-D FEM model of a turbine blade composed of the large 3-D blade root and the shell shaped blade will be used as an example [2].

Numerical Results

A comparison between the new integration methods for the numerical analysis was performed by calculating static stress, natural frequencies, and vibration modes of a rectangular steel plate and bladed disk be means of the program package BLADIS+ [2]. The finite element models comprised 16 and 20 nodes hexahedral finite elements. Table 1 shows the integration methods to be applied.

The investigation of the integration methods for the dynamics analysis was carried out by calculating natural frequencies and vibration modes of a rectangular steel plate. The plate dimensions were as follows: length -0.1524 m; width - 0.0254 m; thickness - 0.00159 m. The natural frequencies of the plate vibrations for different options of integration methods compared with the test and the beam theory are given in Table 2 (B-Bending, T-Torsion).

Figure 1 shows the influence of the finite element (FE) IQTM 48 [2] thickness for the plate on the error of natural frequency calculations by „complete 1" integration. So, if the thickness of the FE is in the limits of 80-90% of the other two dimensions of the FE, then the error in the natural frequencies for first three bending mode shapes is minimal ($\approx 1\%$). So, the dimensions of the FE should not differ by more then 20%.

Table 1

Integration options	Integration rules							
	M	K	K_{dir}	K_{sh}	K_{vol}	K_{dist}	$(1-\beta)K_0$	βK_c
Complete 1	3x3x2	2x2x2						
Complete 2	3x3x2	3x3x3						
Complete 3	3x3x2	4x4x4						
Selectdirect 1	3x3x2		2x2x2	3x3x2				
Selectdirect 2	3x3x2		3x3x2	2x2x2				
Selectvol 3	3x3x2				2x2x2	3x3x2		
Selectvol 4	3x3x2				2x2x2	2x2x2		
Reduced 1	3x3x2	3x3x2						
Reduced 2	3x3x2	4x4x2						
Mixed 1								
$\beta = 0.0001$	3x3x2						2x2x2	3x3x2
$\beta = 0.001$	3x3x2						2x2x2	3x3x2
$\beta = 0.01$	3x3x2						2x2x2	3x3x2
Mixed 2								
$\beta = 0.01$	3x3x2						2x2x2	1x1x1

A disk with a free central hole was considered as an example for calculating the static strength due to centrifugal forces (587.65 rad/s). The disk dimensions were 0.240 m and 0.157 m for the outer and inner radius, respectively with the thickness 0.001m. The calculation was carried out in polar-cylindrical coordinates, and a 30^0 - disk sector was considered. The finite element sector model comprised 20 hexahedral FE IQTM48 with 48 degrees of freedom [2]. The strains obtained at the middle radius compared with analytical results are given in Table 3. This table shows that, in general, all options of selective integration

Table2

Integration option	Natural frequencies (Hz) of rectangular plate			
	1B	2B	1T	1Bmax
Complete 1	56.0	351.9	746.9	871.1
Complete 2	56.6	409.2	803.1	876.1
Complete 3	57.8	417.9	820.3	894.8
Selectdirect 1	56.6	358.4	762.9	889.7
Selectdirect 2	57.8	417.9	820.3	894.8
Selectvol 3	57.8	417.9	820.2	894.8
Selectvol 4	56.6	358.4	762.9	889.6
Reduced 1	57.6	417.1	819.9	894.7
Reduced 2	57.8	417.9	820.3	894.8
Mixed 1				
$\beta = 0.0001$	57.2	385.7	772.9	889.0
$\beta = 0.001$	56.7	363.0	764.1	889.7
$\beta = 0.01$	56.6	358.9	763.0	889.7
Mixed 2				
$\beta = 0.01$	56.6	359.0	762.9	889.6
Beam theory	57	357	692	911
Experiment	60-64	370-398	663-669	-

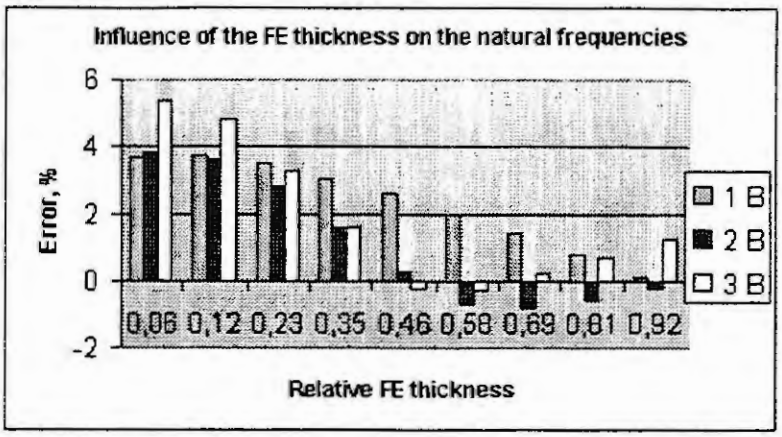


Fig. 1 Influence of the FE thickness of on the natural frequency calculations of a plate

and mixed integration (except mixed $\beta = 0.001$ for radial stress) gave a better accuracy than a complete integration.

Table3

Integration options		Radial stress	Error, %	Tangential stress	Error, %
Complete	1	43.65	-4.7	635.4	-0.9
Selectdirect	2	45.15	-1.23	643.67	0.37
Selectvol	3	45.22	-1.07	638.78	-0.39
Mixed	1				
$\beta = 0.0001$		44.65	-2.32	648.18	1.07
$\beta = 0.001$		42.07	-7.96	636.34	-0.78
$\beta = 0.01$		44.77	-2.06	642.70	0.22
		45.30	-0.89	639.31	-0.31
Mixed	2				
$\beta = 0.01$		45.30	-0.89	639.31	-0.31
Analytic		45.71	-	641.31	-

It is worth note that all integration options (except mixed integration with $\beta = 0.0001$) produced errors less than 1% in the tangential stress. Complete integration gives a significant error for radial stresses at the expense of great shear component that adversely affects the results in our case when the disk thickness is 1 mm, and the elements are therefore very thin. So, in cases when the elements, composing the structure, are thin and deviate largely from the correct shape, it would be appropriate to use the reduced integration order for the acquisition of more accurate results.

Acknowledgments

The research presented in this paper was partly supported by the Austrian Science Foundation (FWF). This is a joint work with I. Ryjikov from Irkutsk State Technical University.

References

1. Kelen, P., A finite element analysis of the vibration characteristics of rotation turbine assemblies: PhD- Thesis, Surrey, 1985.
2. Repetski, O, Computer analysis of dynamics and strength of turbomachines: ISTU, Irkutsk, 1999 (in Russ.).
3. Repetski, O, Numerical Integration in the 3D-Finite Element Analysis In: Proc. of 4th International Congress on Industrial and Applied Mathematics, Edinburgh, 1999, 303.

MODELING OF LINEAR SYSTEMS AND FINITE DETERMINISTIC AUTOMATA BY MEANS OF WALSH FUNCTIONS

U. Konigorski
Clausthal University of Technology
Leibnizstrasse 28, D-38678 Clausthal-Zellerfeld

Abstract. This paper deals with the application of Walsh functions in two different fields of system and control theorie. First of all it is shown that Walsh functions can be used to provide a simple algebraic representation of linear multivariable systems which in turn can be used for controller design. On the other hand, due to the special properties of Walsh functions, they are also especially suited for modeling finite deterministic automata. Applying the approach developed in the paper it will turn out, that any finite deterministic automaton can be mathematically described by a set of first order difference equations. Thus the well known and powerful methods developed for linear discrete-time systems may be applied for analyzing the structure and dynamical behavior of the automaton, e.g. determining dead-locks and cycles.

Introduction

The applications of Walsh functions in system theorie have been given much attention especially in the early seventies. For example, Walsh functions have been applied in the modeling, analysis and time-domain synthesis of linear systems [1] as well as in the analysis and design of communication systems and so-called linear sequency filters [6]. Moreover, in [7] Walsh functions are used for identification while [2] deals with a Walsh series approach to system simulation which has some aspects in common with the one presented in this paper. A comprehensive overview of the application of Walsh functions can be found in [5].

Walsh functions have been first introduced by Walsh [8]. They form a complete orthonormal set of rectangular waves in [0, 1). Therefore, every function $f(t)$ which is absolutely integrable in the interval $0 \leq t < 1$ can be expanded formally in a series of the form

$$f(t) = \sum_{i=0}^{\infty} \widehat{f}_i \cdot \mathrm{wal}_i(t) \tag{1}$$

where $\mathrm{wal}_i(t)$ is the i-th Walsh function and the constants $\widehat{f}_i$ form the *sequency spectrum* of $f(t)$. The series (1) converges uniformly if the terms are grouped so that each group contains all of the Walsh functions designated by m binary digits. If we choose $N = 2^m$ the truncated sum

$$\widetilde{f}(t) = \sum_{i=0}^{N-1} \widehat{f}_i \cdot \mathrm{wal}_i(t) \tag{2}$$

gives an approximation $\widetilde{f}(t)$ of $f(t)$ with minimum integral square error. On the other hand, if $f(t)$ is a stairstep function with N equally spaced subintervals of length 2^{-m} in [0, 1), the partial sum (2) yields an exact representation of $f(t)$. Another quite interesting feature of the Walsh transform $\mathfrak{W}$ is the existence of the so-called dyadic convolution theorem $\mathfrak{W}\{f(t) \cdot g(t)\} = \widehat{f} \circledast \widehat{g}$ where $\circledast$ denotes dyadic multiplication of the corresponding sequency spectra. As will be shown in this paper, these properties of Walsh functions form an adequate basis for modeling linear multivariable systems as well as finite automata.

Algebraic modeling of linear multivariable systems

Following the idea presented in [4], where in contrast to this paper the orthonormal set of Legendre polynomials is used to get an algebraic representation of the state space description

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \quad \mathbf{x}(0) = \mathbf{x}_0 \tag{3}$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \tag{4}$$

of a linear multivariable system with $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{y} \in \mathbb{R}^q$, $\mathbf{u} \in \mathbb{R}^p$ and $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$ constant matrices of appropriate dimensions, we first apply the nonlinear time-transformation $\tau = 1 - e^{-\alpha t}$ to (3),(4) to obtain the transformed

state equations

$$\alpha(1-\tau)\overline{\mathbf{x}}'(\tau) = \mathbf{A}\overline{\mathbf{x}}(\tau) + \mathbf{B}\overline{\mathbf{u}}(\tau), \quad \overline{\mathbf{x}}(0) = \mathbf{x}_0 \tag{5}$$

$$\overline{\mathbf{y}}(\tau) = \mathbf{C}\overline{\mathbf{x}}(\tau) + \mathbf{D}\overline{\mathbf{u}}(\tau) \tag{6}$$

where $'$ denotes differentiation with respect to τ. In (5),(6) $\overline{\mathbf{x}}(\tau)$, $\overline{\mathbf{y}}(\tau)$, $\overline{\mathbf{u}}(\tau)$ are functions of τ, $0 \leq \tau < 1$ and thus we can use the truncated series expansions

$$\overline{\mathbf{x}}(\tau) = \sum_{i=0}^{N-1} \widehat{\mathbf{x}}_i \cdot \mathrm{wal}_i(\tau), \quad \overline{\mathbf{y}}(\tau) = \sum_{i=0}^{N-1} \widehat{\mathbf{y}}_i \cdot \mathrm{wal}_i(\tau), \quad \overline{\mathbf{u}}(\tau) = \sum_{i=0}^{N-1} \widehat{\mathbf{u}}_i \cdot \mathrm{wal}_i(\tau)$$

to get

$$\widehat{\mathbf{X}}\Delta - \mathbf{X}_a = \mathbf{A}\widehat{\mathbf{X}} + \mathbf{B}\widehat{\mathbf{U}} \tag{7}$$

$$\widehat{\mathbf{Y}} = \mathbf{C}\widehat{\mathbf{X}} + \mathbf{D}\widehat{\mathbf{U}} \tag{8}$$

as an algebraic approximation of the system (5),(6) in the sequency domain with $\widehat{\mathbf{X}} = [\widehat{\mathbf{x}}_0, \ldots, \widehat{\mathbf{x}}_{N-1}]$, $\mathbf{X}_a = \alpha \cdot 2^{m+1}[0, \ldots, 0, \mathbf{x}_0]$ and Δ the *Walsh operational matrix for differentiation*. The close relation of (7),(8) with the conventional Laplace transform of (3),(4) becomes even more evident by transforming Δ to diagonal form, $\mathbf{V}^{-1}\Delta\mathbf{V} = \mathbf{V}^{-1}\Delta[\mathbf{v}_0, \ldots, \mathbf{v}_{N-1}] = \mathbf{diag}(\gamma_0, \ldots, \gamma_{N-1})$, to end up with the following simple equations

$$\mathbf{y}_i^* = [\mathbf{C}(\gamma_i\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}]\mathbf{u}_i^* = \mathbf{G}(\gamma_i) \cdot \mathbf{u}_i^*, \quad \gamma_i = \alpha(3+2i), \quad i = 0, ..., N-1 \tag{9}$$

where $\mathbf{y}_i^* = \widehat{\mathbf{Y}}\mathbf{v}_i$, $\mathbf{u}_i^* = \widehat{\mathbf{U}}\mathbf{v}_i$ are the transformed Walsh-Fourier-coefficient vectors (and $\mathbf{x}_0$ is assumed 0 for simplicity). (9) constitutes an ideal basis for analysis and synthesis of linear systems in the sequency domain. For example, fig. (1) shows an approximation to the step response of the third order system $y(s) = \frac{20}{(s+2s+10)(s+2)}u(s) = G(s)u(s)$ which is obtained by using the first 32 Walsh functions. Moreover, (9) can also be used for the design of a standard PI-controller $u(s) = \frac{K_p s+K_I}{s}(w(s) - y(s)) = G_R(s)e(s)$. To that purpose, the desired closed loop response is specified by the transfer function $G_d(s) = \frac{8}{(s+2)(s+2)(s+2)}$ which in turn is set equal to the systems closed loop transfer matrix $G_c(s) = \frac{G_R(s)G(s)}{1+G_R(s)G(s)}$ at the two real points $\gamma_1 = 3\alpha$, $\gamma_2 = 5\alpha$. According to (9) by virtue of this procedure the first two Walsh-Fourier-coefficients of $y_d(s) = G_d(s)w(s)$ and $y_c(s) = G_c(s)w(s)$ are made equal. Finally, with $\alpha = 0.1 \cdot \ln(2)$ the two resulting linear equations yield the solution $K_p = 0.143$, $K_I = 0.665$. As can be seen from fig. (2) applying these controller values there is quite a good match between $y_d(t)$ and $y_c(t)$.

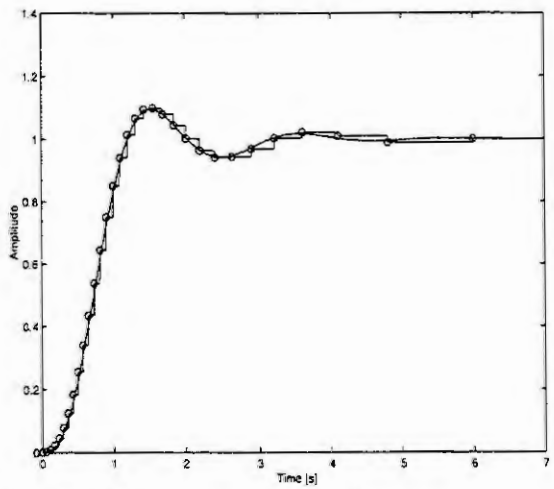


Figure: 1 Walsh series approximation

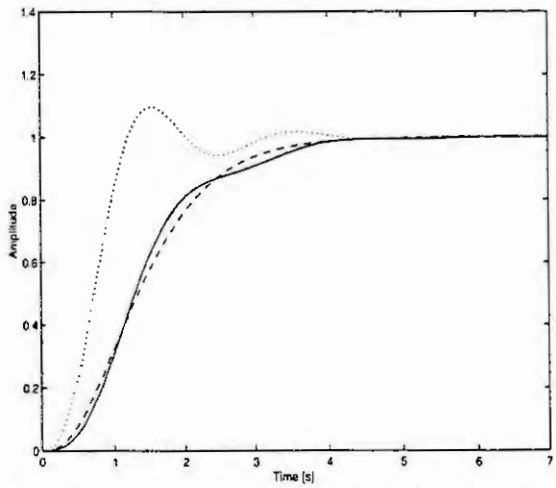


Figure: 2 Step response of $y(t)$ (..), $y_d(t)$ (- -), $y_c(t)$ (–)

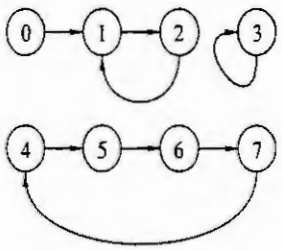
Modeling of finite deterministic automata

In [3] based on the use of Shegalkin polynomials a linear state space approach to a class of discrete-event systems has been proposed . In what follows it will be shown, that an approach similar to the one in the previous section can also be utilized to model finite deterministic automata by the state equation $\mathbf{x}_{k+1} = \mathbf{A} \cdot \mathbf{x}_k$ of a conventional linear

discrete-time system. To that purpose the function f in the state equation $y_{k+1} = f(y_k)$ of a finite automaton is interpreted as a stairstep function over y_k which in turn can exactly be represented by the finite Walsh series

$$f(y_k) = \sum_{i=0}^{N-1} \hat{f}_i \cdot \mathrm{wal}_i(y_k) \tag{10}$$

with $N = 2^m$ Walsh-Fourier-coefficients, if the automaton has less than or equal N individual states y_k which can be associated with the integer numbers $\{0, 1, \ldots, N-1\}$ as illustrated in the example below.



If y_k and y_{k+1} are represented in binary notation

	y_k			y_{k+1}
000	0		001	1
001	1		010	2
010	2		001	1
011	3	$\overset{f}{\longmapsto}$	011	3
100	4		101	5
101	5		110	6
110	6		111	7
111	7		100	4

and by substituting $0 \mapsto 1$, $1 \mapsto -1$ we get the relations

$$\begin{array}{lll} x_{1k} = \mathrm{wal}_1(y_k) & & x_{1k+1} = \mathrm{wal}_1(y_{k+1}) = \mathrm{wal}_1(f(y_k)) \\ x_{3k} = \mathrm{wal}_3(y_k) & \text{and} & x_{3k+1} = \mathrm{wal}_3(y_{k+1}) = \mathrm{wal}_3(f(y_k)) \\ x_{7k} = \mathrm{wal}_7(y_k) & & x_{7k+1} = \mathrm{wal}_7(y_{k+1}) = \mathrm{wal}_7(f(y_k)) \end{array} \tag{11}$$

as can be seen from the table below.

x_{1k}	x_{3k}	x_{7k}	y_k		x_{1k+1}	x_{3k+1}	x_{7k+1}	y_{k+1}
1	1	1	0		1	1	−1	1
1	1	−1	1		1	−1	1	2
1	−1	1	2		1	1	−1	1
1	−1	−1	3	$\overset{f}{\longmapsto}$	1	−1	−1	3
−1	1	1	4		−1	1	−1	5
−1	1	−1	5		−1	−1	1	6
−1	−1	1	6		−1	−1	−1	7
−1	−1	−1	7		−1	1	1	4

Obviously x_{1k+1}, x_{3k+1}, x_{7k+1} are stairstep functions over y_k and thus can be exactly expressed by their corresponding Walsh series, e.g.

$$\begin{aligned} x_{3k+1} &= \textstyle\sum_{i=0}^{7} \hat{x}_{3i} \cdot \mathrm{wal}_i(y_k) \\ &= 0.5\mathrm{wal}_4(y_k) - 0.5\mathrm{wal}_5(y_k) + 0.5\mathrm{wal}_6(y_k) + 0.5\mathrm{wal}_7(y_k) \\ &= 0.5\underbrace{\mathrm{wal}_4(y_k)}_{:=x_{4k}} - 0.5\underbrace{\mathrm{wal}_5(y_k)}_{:=x_{5k}} + 0.5\underbrace{\mathrm{wal}_6(y_k)}_{:=x_{6k}} + 0.5x_{7k} \end{aligned} \tag{12}$$

Next the three Walsh functions $\mathrm{wal}_4(y_k)$, $\mathrm{wal}_5(y_k)$, $\mathrm{wal}_6(y_k)$ appearing in (12) are defined as additional state variables x_{4k}, x_{5k} x_{6k} and after applying the state transition function $f(y_k)$ the resulting stairstep functions

$x_{ik+1} = \mathrm{wal}_i(f(y_k)),\ i = 4,5,6$ are again expressed by their corresponding finite Walsh series

$$x_{4k+1} = \mathrm{wal}_4(y_{k+1}) = \mathrm{wal}_4(f(y_k)) = \sum_{i=0}^{7} \widehat{x}_{4i} \cdot \mathrm{wal}_i(y_k)$$
$$x_{5k+1} = \mathrm{wal}_5(y_{k+1}) = \mathrm{wal}_5(f(y_k)) = \sum_{i=0}^{7} \widehat{x}_{5i} \cdot \mathrm{wal}_i(y_k)$$
$$x_{6k+1} = \mathrm{wal}_6(y_{k+1}) = \mathrm{wal}_6(f(y_k)) = \sum_{i=0}^{7} \widehat{x}_{6i} \cdot \mathrm{wal}_i(y_k)$$

Following this procedure we finally end up with the desired linear difference equation

$$\begin{bmatrix} x_{0k+1} \\ x_{1k+1} \\ \vdots \\ x_{7k+1} \end{bmatrix} = \begin{bmatrix} \widehat{x}_{00} & \widehat{x}_{01} & \cdots & \widehat{x}_{07} \\ \widehat{x}_{10} & \widehat{x}_{11} & \cdots & \widehat{x}_{17} \\ \vdots & \vdots & \ddots & \vdots \\ \widehat{x}_{70} & \widehat{x}_{71} & \cdots & \widehat{x}_{77} \end{bmatrix} \mathbf{x}_k$$

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k \tag{13}$$

of order up to $N = 2^m$. Therefore, each finite deterministic automaton can be modeled by a linear difference equation, which offers new possibilities in analyzing such types of automata by means of the well known methods developed for linear discrete-time systems. In the example above, A has the 8 eigenvalues $\Lambda = \{-1, -1, 0, \pm j, 1, 1, 1\}$, where the eigenvalues -1 and $\pm j$ correspond to cycles of length 2 and 4 respectively while an eigenvalue of 1 indicates a steady state or dead-lock of the automaton. Finally, the eigenvalue 0 corresponds to the chain $0 \mapsto 1$. However, as can be seen from the above list of eigenvalues, not every single eigenvalue corresponds to a specific cycle. In general, if r indicates the length of a cycle, where a dead-lock corresponds to a cycle of length 1, it can be easily verified, that the corresponding states are solutions of the linear equation $(\mathbf{I} - \mathbf{A}^r)\mathbf{x}_k = 0$, that is they are the corresponding eigenvectors to the eigenvalues 1 of $\mathbf{A}^r$. Thus the individual states $\mathbf{x}_k$ which constitute a specific cycle of length r of the automaton can be directly obtained from this equation. On the other hand, since the state vectors $\mathbf{x}_k = [\mathrm{wal}_1(y_k), \ldots, \mathrm{wal}_N(y_k)]^T$ are known for each value of k, the above linear equation can also be used to check whether a specific $\mathbf{x}_k$ belongs to a cycle of length r.

Conclusion

It has been shown, that Walsh functions can preferably be used in the modeling and design of linear multivariable systems as well as in the field of discrete event systems. They allow for a simple and efficient design procedure for linear multivariable systems and due to their inherent properties they yield a simple linear difference equation (13) for finite deterministic automata. This also includes some kinds of petri-nets, e.g. so-called condition/event nets. Based upon (13) dead-locks and cycles of such automata can easily be determined. Moreover, a detailed analysis of the structure of the system matrix A provides results concerning the structure of the underlying automaton.

References

[1] Chen, C.F. and Hsiao, C.H., Time-domain synthesis via Walsh functions. IEE Proc., Vol 122, 1975, pp. 565-570.

[2] Chen, C.F. and Hsiao, C.H., A state space approach to Walsh series solution of linear systems. Int. J. System Sci., 1975, Vol. 6, No. 9, pp. 833-858.

[3] Franke, D., A linear state space approach to a class of discrete-event systems. Mathematics and Computers in Simulation, Vol. 39, 1995, pp. 499-503.

[4] Franke, D., Krüger, K., and Knoop, M., Systemdynamik und Reglerentwurf. R. Oldenbourg Verlag, München, 1992.

[5] Harmuth, H.F., Sequency theory. Academic Press, New York, 1970.

[6] Pichler, F., Synthese linearer periodisch zeitvariabler Filter mit vorgeschriebenem Sequenzverhalten. A.E.Ü., Band 22, 1968, pp. 150-161.

[7] Prasada Rao, G., and Sivakumar, L., System identification via Walsh functions. IEE Proc., Vol 122, 1975, pp. 1160-1161.

[8] Walsh, J.L., A closed set of normal orthogonal functions. Amer. J. Math., Vol 45, 1923, pp. 5-24.

COMPUTING CHRISTOFFEL SYMBOLS FOR MODELING WITH PDEs ON CONFORMAL GRIDS

M. Holzinger, F. Breitenecker, H.J. Dirschmid
Vienna University of Technology
Wiedner Hauptstraße 8–10, A–1040 Vienna, Austria

Abstract. Modeling and simulation with partial differential equations using finite differences on complex domains usually require a special transformation of the given physical plane parametrized by means of cartesian coordinates onto a computational plane maintaining the property of orthogonality. Introducing conformal mappings — which are preserving angles — one achieves such simplifications whereas on the other hand the problem of an invariant formulation of the governing equations arises. This contribution presents some results of the numerical computation of Christoffel symbols of second kind on a star-shaped manifold. These quantities are derived from the fundamental tensor and are acting as correction terms when covariant differentiation is used. The Einstein summation convention is applied throughout this paper.

Introduction

We start our investigations on an arbitrary two-dimensional compact, connected and flat Riemannian manifold $\mathcal{M}$ with the only restriction that it should be star-shaped in respect to the origin, which in terms of parametrization by means of a chart $\kappa : \mathbb{R}^2 \to \mathcal{M}$ can be stated as $\kappa(x_0 + t(x_i - x_0)) \in \mathcal{M} \quad \forall t \in [0,1], \quad \forall X \in \mathcal{M}$. The boundary of $\mathcal{M}$ will be denoted by a closed Jordan arc C. In [4] we demonstrated how to construct a conformal grid by means of solving Theodorsens nonlinear and singular integral equation which yields a function of boundary correspondence, $\theta(\varphi)$, relating the angle φ on the unit disk with the corresponding angle θ on C. Subsequent Fourier expansion of $\ln \varrho(\theta(\varphi))$ — with ϱ denoting the radius $|OX|, X \in C$ — and the fact of a connection between the real and imaginary parts of the points on the unit disk to be mapped via conjugated functions finally resulted in finite sums

$$\begin{aligned} \varrho(r,\varphi) &= r \exp\left\{ \frac{a_0}{2} + \sum_{k=1}^{N-1} r^k (a_k \cos k\varphi + b_k \sin k\varphi) + r^N \frac{a_N}{2} \cos N\varphi \right\} \\ \theta(r,\varphi) &= \varphi + \sum_{k=1}^{N-1} r^k (a_k \sin k\varphi - b_k \cos k\varphi) + r^N \frac{a_N}{2} \sin N\varphi, \end{aligned} \tag{1}$$

which should for the following be interpreted as an approximation for a transformation of coordinates. The radius ϱ was provided for any desired angle θ by means of cubic non-parametric spline-interpolation of a given set of points describing the boundary C. The computational grid is shown in figure (1).

The Metric Coefficients

The coordinates of the covariant metric tensor[1] in an arbitrary point, $P \in \mathcal{M}$, with reference to local basis vectors depending on the parametrization of the manifold are given by the inner products

$$g_{ij}(P) := (\partial_i, \partial_j)(P), \quad i,j = 1,2.$$

In case of cartesian coordinates, these expressions simply reduce to the constant 2×2-unit matrix for each point on the manifold. Changing the parametrization of $\mathcal{M}$ affects the metric tensor by the cogredient transformation law

$$\bar{g}_{ij}(P) = \frac{\partial x_k}{\partial \bar{x}_i} \frac{\partial x_l}{\partial \bar{x}_j} g_{kl}(P). \tag{2}$$

In case of switching from cartesian to polar coordinates which is performed by functions $\xi(\varrho,\theta) = \varrho\cos\theta, \quad \eta(\varrho,\theta) = \varrho\sin\theta,$

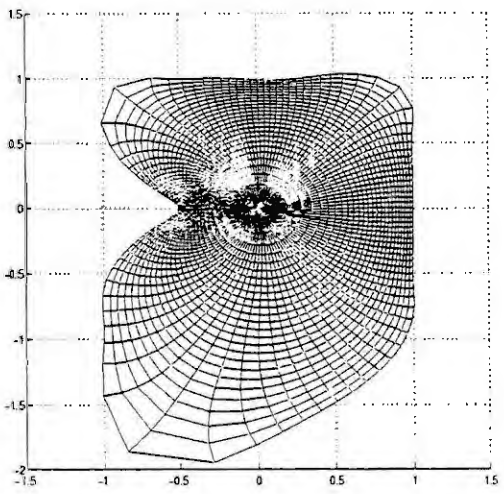


Figure 1: Conformal grid as result of a map from the unit-disk onto a star-shaped geometry.

one easily obtains $g_{11} = 1$, $g_{22} = \varrho^2$ and zero non-diagonal elements which indicates the preservation of orthogonality. Note that for $\varrho = 0$, there is a coordinate-singularity caused by a vanishing determinant of the fundamental tensor. Applying (2) in connection with functions (1) — which in fact yields the desired parametrization of the boundary C by means of $(1, \varphi)$, $0 \leq \varphi < 2\pi$, and thus the computational grid by mapping the unit disk conformally onto the geometry — the covariant metric tensor in computational coordinates (r, φ) finally is of the form

$$g_{ij}(r,\varphi) = \begin{pmatrix} \varrho_r^2 + \varrho^2\theta_r^2 & 0 \\ 0 & \varrho_\varphi^2 + \varrho^2\theta_\varphi^2 \end{pmatrix}.$$

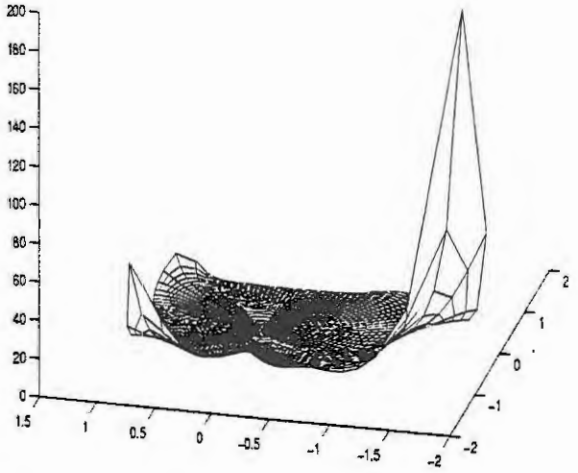


Figure 2: Coordinate g_{11} of the fundamental tensor.

Again orthogonality of the basis vectors — which are for the latter transformation even stretched or shrinked by the same quantity by the fact that the Cauchy-Riemannian differential equations hold for conformal mappings — is retained. Moreover, applying both the transformations in reverse order, one obtains $g_{11} = \chi$ and $g_{22} = r^2\chi$ with $\chi = \xi_x^2 + \eta_x^2$ defined as the absolute value of the conformal mapping which points out that the non-vanishing coordinates are proportional. Figure (2) therefore only shows coordinate g_{11}.
Once the metric tensor is known, one derives the Christoffel symbols of second kind from [1]

$$\Gamma_{jk}^i = \frac{1}{2} g^{il} \left(\frac{\partial g_{lj}}{\partial x_k} + \frac{\partial g_{lk}}{\partial x_j} - \frac{\partial g_{jk}}{\partial x_l} \right), \tag{3}$$

with the contravariant fundamental tensor g^{ij} appearing in the summation above defined by relations $g^{ij} g_{jk} = \delta_k^i$, which in our case means $g^{ii} = g_{ii}^{-1}$. Taking symmetries of the symbols in the lower indices into account — one concludes this from the fact that affine connections in Riemann spaces are free of torsion — and resolving the summation, equation (3) provides 6 relevant terms,

$$\begin{aligned} \Gamma_{11}^1 &= \frac{1}{2} g^{11} \frac{\partial g_{11}}{\partial r} & \Gamma_{12}^1 &= \frac{1}{2} g^{11} \frac{\partial g_{11}}{\partial \varphi} & \Gamma_{22}^1 &= -\frac{1}{2} g^{11} \frac{\partial g_{22}}{\partial r} \\ \Gamma_{12}^2 &= \frac{1}{2} g^{22} \frac{\partial g_{22}}{\partial r} & \Gamma_{22}^2 &= \frac{1}{2} g^{22} \frac{\partial g_{22}}{\partial \varphi} & \Gamma_{11}^2 &= -\frac{1}{2} g^{22} \frac{\partial g_{11}}{\partial \varphi}. \end{aligned} \tag{4}$$

Moreover, when reconsidering the proportionality of g_{11} and g_{22} mentioned above, one finds that in case of azimuthal derivative the radius cancels down and hence additionally is able to show $\Gamma_{22}^2 = \Gamma_{12}^1$.

Approximating the Christoffel symbols

We first state the possibility to numerically provide the metric tensor for an arbitrary point of $\mathcal{M}$ immediately — compare again for Figure (2) — by differentiating the finite series in (1) which yields to

$$\begin{aligned} \varrho_r &= \varrho \left[\frac{1}{r} + \sum_{k=1}^{N-1} k r^{k-1} (a_k \cos k\varphi + b_k \sin k\varphi) + N r^{N-1} \frac{a_N}{2} \cos N\varphi \right] \\ \varrho_\varphi &= \varrho \left[\sum_{k=1}^{N-1} k r^k (b_k \cos k\varphi - a_k \sin k\varphi) - N r^N \frac{a_N}{2} \sin N\varphi \right] \\ \theta_r &= \sum_{k=1}^{N-1} k r^{k-1} (a_k \sin k\varphi - b_k \cos k\varphi) + N r^{N-1} \frac{a_N}{2} \sin N\varphi \\ \theta_\varphi &= 1 + \sum_{k=1}^{N-1} k r^k (a_k \cos k\varphi + b_k \sin k\varphi) + N r^N \frac{a_N}{2} \cos N\varphi. \end{aligned}$$

In principle, one can proceed in an analogous manner to evaluate the Cristoffel symbols by computing the second derivatives. However, usage of boundary-interpolation with cubic periodic splines on the one hand results in a lack of smoothness for the second derivatives. On the other hand, evaluation of the Fourier series in points which were not used as interpolation points gives rise to undesirable oscillations especially at the boundary. To avoid these problems one should better make use of finite-difference schemes. On non-equidistant grids,the first derivative of a state variable u with respect to an independent spatial variable can be approximated by [5]

$$\left.\frac{\partial u}{\partial x}\right|_i \approx u_{i+1}\frac{\Delta_{i-1}}{\Delta_i\left(\Delta_i+\Delta_{i-1}\right)}+u_i\left(\frac{1}{\Delta_{i-1}}-\frac{1}{\Delta_i}\right)-u_{i-1}\frac{\Delta_i}{\Delta_{i-1}\left(\Delta_i+\Delta_{i-1}\right)}$$

for all interior grid points with Δ_i denoting the distance to an outward or upward neighbor point respectively. At the boundary an inward differentiation formula [2] like

$$\left.\frac{\partial u}{\partial x}\right|_1 \approx -u_3\frac{\Delta_1}{\Delta_2\left(\Delta_1+\Delta_2\right)}+u_2\left(\frac{1}{\Delta_1}+\frac{1}{\Delta_2}\right)-u_1\left(\frac{1}{\Delta_1}+\frac{1}{\Delta_1+\Delta_2}\right),$$

can be applied. It is important to note that, caused by the singularity in the origin, all grid-points on the inner computational circle have also been treated in this way. Furthermore, as a closer look on equation (4) reveals, in some cases it is possible to make use of the logarithm — e.g. $2\Gamma^1_{11}=\partial \ln g_{11}/\partial r$ holds — before the discretization process.

Conclusion

Covariant differentiation of tensor fields of arbitrary rank is defined as

$$\frac{\mathfrak{d}\Phi^{i_1\cdots i_n}_{j_1\cdots j_m}}{\mathfrak{d}x_p}:=\frac{\partial\Phi^{i_1\cdots i_n}_{j_1\cdots j_m}}{\partial x_p}+\sum_{h=1}^{n}\Gamma^{i_h}_{lp}\Phi^{i_1\cdots l\cdots i_n}_{j_1\cdots j_m}-\sum_{k=1}^{m}\Gamma^{l}_{j_k p}\Phi^{i_1\cdots i_n}_{j_1\cdots l\cdots j_m}.$$

Whenever such expressions have to be computed, one is therefore able to do so by approximating the conventional spatial derivatives and subsequently adding or subtracting products formed by Christoffel symbols and the state variable itself. These terms can be interpreted as to take the local (non-Euclidean) geometrical conditions into account and depend on the co- or contravariant nature of the field variable. Whilst for scalar fields covariant and ordinary spatial differentiation coincide, the divergence of a contravariant vector field for example has to be corrected, that is

$$\operatorname{div} v=\frac{\mathfrak{d}v^i}{\mathfrak{d}x_i}=\frac{\partial v^r}{\partial r}+\Gamma^1_{11}v^r+\Gamma^1_{12}v^\varphi+\frac{\partial v^\varphi}{\partial\varphi}+\Gamma^2_{12}v^r+\Gamma^2_{22}v^\varphi.$$

Figures (3-7) show the computed Christoffel symbols of second kind on $\mathcal{M}$. The similar shapes of Γ^1_{11} and Γ^2_{12} result from the fact that g_{11} and g_{22} are proportional, whereas near the origin the singularity enters in case of Γ^2_{12}.

References

[1] Dirschmid, H.J., Tensoren und Felder. Springer-Verlag, Wien, 1996, ISBN 3-211-82754-4.

[2] Ferziger, J.H. and Perić, M., Computational Research for Fluid Dynamics. Springer-Verlag, Berlin-Heidelberg, 1996, ISBN 3-540-59434-5.

[3] Gaier, D., Konstruktive Methoden der konformen Abbildung. Springer-Verlag, Berlin-Göttingen-Heidelberg, 1964.

[4] Holzinger, M., Breitenecker, F. and Dirschmid, H.J., Conformal Mappings for Simplifying PDE-based Modeling and Simulation. In: Simulation-Past, Present and Future. 12th European Simulation Multiconference, June 16-19, 1998, (Eds.: Zobel,R. and Moeller, D.J.) Manchester, UK, 208-210.

[5] Kantorowitsch, L.W. and Krylow, W.I., Näherungsmethoden der höheren Analysis. VEB, Deutscher Verlag der Wissenschaften, Berlin 1956.

[6] Kythe, P.K., Computational Conformal Mapping. Birkhäuser, Boston 1998, ISBN 0-8176-3996-9.

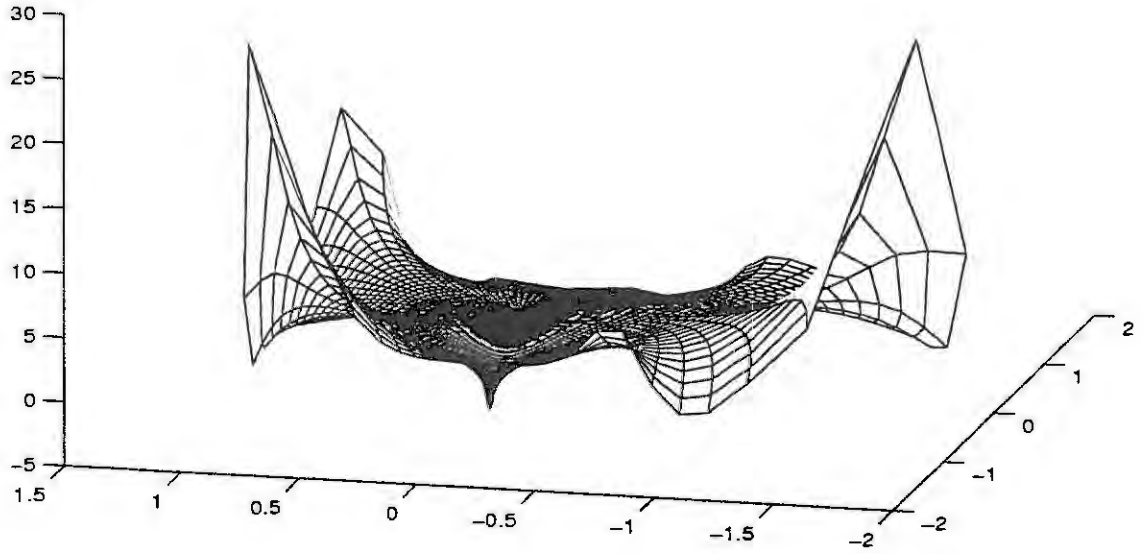


Figure 3: Christoffel symbol of second kind Γ^1_{11}.

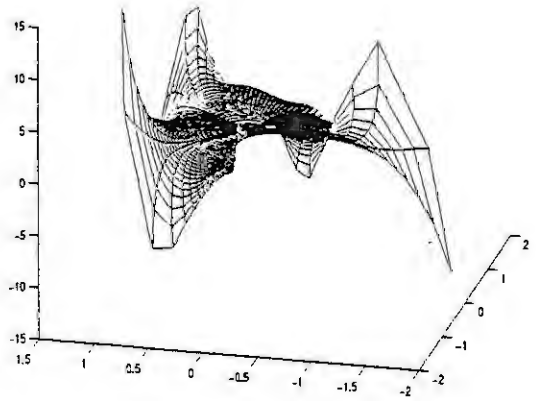


Figure 4: Christoffel symbol of second kind Γ^1_{12}.

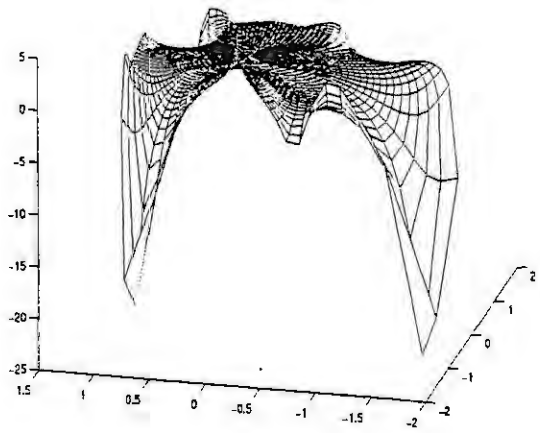


Figure 5: Christoffel symbol of second kind Γ^1_{22}.

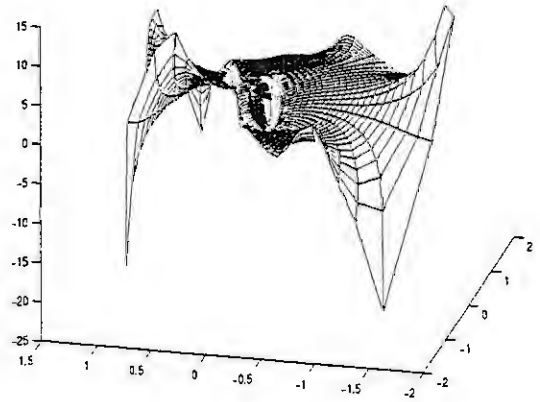


Figure 6: Christoffel symbol of second kind Γ^2_{11}.

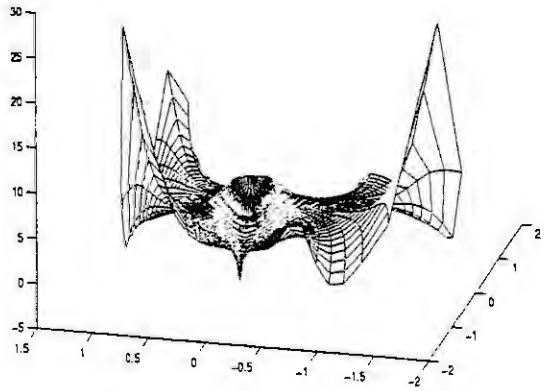


Figure 7: Christoffel symbol of second kind Γ^2_{12}.

ON THE IDENTIFICATION OF NONLINEAR SYSTEMS BY COMBINING IDENTIFIED LINEAR MODELS

D.J.Leith, W.E.Leithead
Department of Electronic & Electrical Engineering, University of Strathclyde,
50 George St., Glasgow G1 1QE, U.K.
Tel. +44 141 548 2407, Fax. +44 141 548 4203, Email. doug@icu.strath.ac.uk

Abstract
Divide and conquer identification approaches are considered with the aim of permitting well-developed linear methods can be brought to bear on the nonlinear identification task. Transfer functions of the plant linearisations are identified from measured data and the requirement is then to infer the underlying nonlinear system. It is shown that knowledge only of the transfer functions of the linearisations of the nonlinear system is insufficient to permit such reconstruction and sufficient conditions based on augmented transfer function knowledge are derived.

1. Introduction

Mathematical models of dynamic systems are required in a wide range of applications. These models may be determined directly from measured experimental data (when, for example, the expense of developing a detailed analytic model cannot be justified), derived analytically from first principles or, perhaps most commonly, determined by some combination of empirical and analytic methods. It should be noted that even in the case of models derived purely by analytic methods, experimental external validation is required in order to establish their accuracy and range of applicability. The identification of linear systems from measured experimental data has received considerable attention over the last thirty years and there exists a wealth of theoretical results relating to issues such as structure identification, parameter estimation, experiment design and model validation testing together with a great deal of accumulated practical experience. However, all systems are in reality nonlinear and identification techniques are less well developed for systems which cannot be accurately approximated by a single linear time-invariant system. This type of situation exists not only in the identification field but also more generally. Whilst nonlinear dynamic systems are widespread, the analysis and design of such systems remains relatively difficult. In contrast, although systems with genuinely linear time-invariant dynamics do not, in reality, exist, techniques for the analysis and design of linear time-invariant systems are rather better developed. It is, therefore, often attractive to consider a divide and conquer strategy whereby the analysis/design of a nonlinear system is decomposed into the analysis/design of a collection of linear time-invariant systems. In the context of control system analysis and design, this type of strategy is well established and forms the basis, for example, of one of the most widely, and successfully, applied techniques for the design of nonlinear controllers; namely, gain-scheduling. Similarly, in the context of system identification it is common practice, when faced with the task of modelling a nonlinear system, to initially identify a number of linear approximations to the system each of which is locally valid.

Traditionally, the first-order Taylor series expansion of a nonlinear system is often employed as a local linear approximation. However, since the first-order expansion is linear only when the expansion is carried out relative to an equilibrium point, consideration is necessarily confined to near equilibrium operation. Of course, while the dynamics in the vicinity of a single equilibrium point are clearly important, the dynamic behaviour during rapid transitions between equilibrium points and, indeed, the behaviour during sustained operation far from equilibrium are also frequently of considerable interest. Fortunately, this issue is addressed by a recent generalisation of the conventional equilibrium linearisation, namely the velocity-based linearisation (Leith & Leithead 1998a,b). In contrast to the conventional series expansion linearisation approach, the velocity-based approach associates a linear system, namely the velocity-based linearisation, with *every* operating point of a nonlinear system (including those far from equilibrium) not just the equilibrium operating points. The solution to the velocity-based linearisation associated with an operating point locally approximates the solution of the nonlinear system and the global solution to the nonlinear system can be recovered by appropriately piecing together the solutions to its velocity-based linearisations. While maintaining continuity with linear methods, the velocity-based approach removes the restriction to near equilibrium operation which is inherent to conventional linearisation approaches and accommodates, for example, both transitions between equilibrium operating points and sustained operation far from equilibrium. Since it describes the dynamics at every operating point, the velocity-based linearisation family associated with a nonlinear system is alternative representation of the system and involves no loss of information.

Whilst originally derived in the context of control system analysis and design, the velocity-based linearisation provides a natural framework within which to consider divide and conquer identification approaches whereby well-developed linear methods can be brought to bear on the nonlinear identification task. In the context of system identification, the transfer functions of the velocity-based linearisations might be identified from measured data using well-established linear methods and the requirement is then to infer the velocity-based linearisation family or, equivalently, the nonlinear system. It is the latter task which is the subject of the present paper. .

2. Velocity-based linearisation

The velocity-based analysis and design representation is briefly summarised. Consider a nonlinear system

$$\dot{x} = F(x, r), \quad y = G(x, r) \tag{1}$$

where $F(\cdot,\cdot)$ and $G(\cdot,\cdot)$ are differentiable nonlinear functions and $r \in \Re^m$ denotes the input to the plant, $y \in \Re^p$ the output and $x \in \Re^n$ the states. Differentiating (1), an alternative representation of the nonlinear system is

$$\dot{x} = w, \quad \dot{w} = \nabla_x F(x, r)w + \nabla_r F(x, r)\dot{r}, \quad \dot{y} = \nabla_x G(x, r)w + \nabla_r G(x, r)\dot{r} \tag{2}$$

The velocity-based formulation, (2), is dynamically equivalent to (1) in the sense that, for appropriate initial conditions, they have the same solution, x. It can be shown (Leith & Leithead 1998a) that the solution $\hat{x}$ to the linear system (the "velocity-based linearisation")

$$\dot{\hat{x}} = \hat{w}, \dot{\hat{w}} = \nabla_x F(x_1, r_1)\hat{w} + \nabla_r F(x_1, r_1)\dot{r}, \quad \dot{\hat{y}} = \nabla_x G(x_1, r_1)\hat{w} + \nabla_r G(x_1, r_1)\dot{r} \tag{3}$$

approximates the solution x to the nonlinear system locally to the operating point (x_1, r_1). Since a linear system (3) is associated with every operating point of the nonlinear system, there is a family of velocity-based linearisations associated with the nonlinear system. Whilst the solution to a single velocity-based linearisation is only a local approximation to the solution of the nonlinear system, the solutions to the members of this family can be pieced together to recover the solution of the nonlinear system. . The direct relationship between the linearisation, (3), and the velocity-based nonlinear system, (2), is clear; namely, the velocity-based linearisation is obtained by simply "freezing" (2) at the relevant operating point.

3. Conventional transfer function knowledge alone is insufficient

Inferring the velocity-based linearisation family from the corresponding family of transfer functions is not quite as straightforward as might at first appear. Some indication of this might be evident from the observation that, although it is common practice to identify the equilibrium linearisations of a nonlinear system, rarely is there any attempt to then combine these linearisations in a rigorous manner in order to recover a description of the nonlinear dynamics as the system moves from the vicinity of one equilibrium point to the vicinity of another. The crux of the problem is that only input-output data is available and so there exist infinitely many choices of state-space realisation of each identified transfer function. When piecing together the solutions to the linear systems in order to recover the solutions to the nonlinear system, this piecing together is carried out in state-space. It is therefore necessary to determine the appropriate choice of state for each linear system which ensures compatibility with the underlying nonlinear system.

That knowledge only of the transfer functions of the velocity-based linearisations is insufficient to enable the underlying nonlinear system to be recovered uniquely can be seen from the following analysis. The nonlinear system (1) has, at the operating point (x_1, r_1), the velocity-based linearisation **Error! Reference source not found.**-(3). Of course, the dynamics of a linear system are invariant under a non-singular state-transformation. Consider, therefore, the nonlinear system for which the velocity-based linearisation, at the operating point (x_1, r_1), is

$$\dot{\hat{\chi}} = T(x_1, r_1)\hat{\omega}, \quad \dot{\hat{\omega}} = T^{-1}(x_1, r_1)\nabla_x F(x_1, r_1)T(x_1, r_1)\hat{\omega} + T^{-1}(x_1, r_1)\nabla_r F(x_1, r_1)\dot{r} \tag{4}$$

$$\hat{v} = \nabla_x G(x_1, r_1)T(x_1, r_1)\hat{\omega} + \nabla_r G(x_1, r_1)\dot{r} \tag{5}$$

where $\chi, \omega, \hat{\chi}, \hat{\omega} \in \Re^n$ and $T(\bullet, \bullet)$ is a uniformly bounded non-singular matrix which is differentiable with uniformly bounded derivatives. It can be seen that the velocity-based linearisations, **Error! Reference source not found.**-(3) and (4)-(2), are related by the non-singular transformation $\hat{\omega} = T^{-1}(x_1, r_1)\hat{w}, \quad \hat{\chi} = \hat{x}, \quad \hat{v} = \hat{y}$ (6)

and so are dynamically equivalent. Similarly for the velocity-based linearisations at other operating points. The velocity-form of the nonlinear system with linearisation family (4)-(2) is

$$\dot{\chi} = T(\chi, r)\omega, \quad \dot{\omega} = T^{-1}(\chi, r)\nabla_x F(\chi, r)T(\chi, r)\omega + T^{-1}(\chi, r)\nabla_r F(\chi, r)\dot{r} \tag{7}$$

$$\dot{v} = \nabla_x G(\chi, r)T(\chi, r)\omega + \nabla_r G(\chi, r)\dot{r} \tag{8}$$

Letting $\omega = \mathbf{T}^{-1}(\chi, r)z$ the nonlinear system, (7)-(8), may be reformulated as

$$\dot{\chi} = z, \quad \dot{z} = \nabla_x F(\chi, r)z + \nabla_r F(\chi, r)\dot{r} + \varepsilon \tag{9}$$

$$\dot{v} = \nabla_x G(\chi, r)z + \nabla_r G(\chi, r)\dot{r} \tag{10}$$

where $\varepsilon = \dot{\mathbf{T}}(\chi, r)\,\mathbf{T}^{-1}(\chi, r)z$. Despite the dynamic equivalence of the members of the velocity-based linearisation families, it is evident that the dynamics of the nonlinear systems, **Error! Reference source not found.**-(2) (equivalently (1)) and (7)-(8), are *not* the same. The difference between the dynamics is embodied by the perturbation term, ε, and arises from the variation of the state transformation, (12), with the operating point.

4. Conditions for Reconstructing a Nonlinear System from its Identified Linearisations

It is evident from the foregoing analysis that additional information is required in order to permit the nonlinear system associated with a family of identified transfer functions to be reconstructed. Various types of information might, of course, provide the required additional information but consideration here is confined to a simple, but effective, extension of the available transfer function information.

Before proceeding, it is useful to reformulate the nonlinear system, (1), as

$$\dot{x} = \mathbf{A}x + \mathbf{B}r + \mathbf{f}(\rho), \quad y = \mathbf{C}x + \mathbf{D}r + \mathbf{g}(\rho) \tag{11}$$

where $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$ are appropriately dimensioned constant matrices, $f(\bullet)$ and $g(\bullet)$ are nonlinear functions and $\rho(x,r) \in \Re^q$, $q \le m+n$, embodies the nonlinear dependence of the dynamics on the state and input with $\nabla_x\rho$, $\nabla_r\rho$ constant. Trivially, this reformulation can always be achieved by letting $\rho = [x^T \;\; r^T]^T$, in which case $q=m+n$. However, the nonlinearity of the system is frequently dependent on only a subset of the states and inputs, in which case the dimension, q, of ρ is less than $m+n$.

Consider two nonlinear systems

$$\dot{x} = \mathbf{A}x + \mathbf{B}r + \mathbf{f}(\rho), \quad y = \mathbf{C}x + \mathbf{D}r + \mathbf{g}(\rho) \tag{12}$$

and

$$\dot{\tilde{x}} = \tilde{\mathbf{A}}\tilde{x} + \tilde{\mathbf{B}}r + \tilde{f}(\tilde{\rho}), \quad \tilde{y} = \tilde{\mathbf{C}}\tilde{x} + \tilde{\mathbf{D}}r + \tilde{g}(\tilde{\rho}) \tag{13}$$

where $r \in \Re^m$, $w, \tilde{w} \in \Re^n$, $y, \tilde{y} \in \Re^p$, $\rho, \tilde{\rho} \in \Re^q$. Differentiate to obtain the corresponding velocity-based nonlinear systems

$$\begin{aligned} \dot{\rho} &= \nabla_x\rho\, w + \nabla_r\rho\, r, \dot{w} = (\mathbf{A} + \nabla\mathbf{f}(\rho)\nabla_x\rho)w + (\mathbf{B} + \nabla\mathbf{f}(\rho)\nabla_r\rho)\dot{r} \\ \dot{y} &= (\mathbf{C} + \nabla\mathbf{g}(\rho)\nabla_x\rho)w + (\mathbf{D} + \nabla\mathbf{g}(\rho)\nabla_r\rho)\dot{r} \end{aligned} \tag{14}$$

and

$$\begin{aligned} \dot{\tilde{\rho}} &= \nabla_{\tilde{x}}\tilde{\rho}\,\tilde{w} + \nabla_r\tilde{\rho}\,\dot{r}, \dot{\tilde{w}} = (\tilde{\mathbf{A}} + \nabla\tilde{\mathbf{f}}(\tilde{\rho})\nabla_{\tilde{x}}\tilde{\rho})\tilde{w} + (\tilde{\mathbf{B}} + \nabla\tilde{\mathbf{f}}(\tilde{\rho})\nabla_r\tilde{\rho})\dot{r} \\ \dot{\tilde{y}} &= (\tilde{\mathbf{C}} + \nabla\tilde{\mathbf{g}}(\tilde{\rho})\nabla_{\tilde{x}}\tilde{\rho})\tilde{w} + (\tilde{\mathbf{D}} + \nabla\tilde{\mathbf{g}}(\tilde{\rho})\nabla_r\tilde{\rho})\dot{r} \end{aligned} \tag{15}$$

The members of the velocity-based linearisation families are obtained by "freezing" the velocity-based nonlinear systems (14) and (15). Assume that the nonlinear systems are minimal representations in the sense that (1) the members of the velocity-based linearisations families are controllable and observable, (2) the pairs $(\mathbf{A},\mathbf{C})$ and $(\tilde{\mathbf{A}}, \tilde{\mathbf{C}})$ are observable, and (3) $\nabla f(0)$, $\nabla\tilde{f}(0)$, $\nabla g(0)$, $\nabla\tilde{g}(0)$ are equal to zero. Condition 1 is a standard minimality condition from linear theory whilst conditions 2 and 3 remove the possible ambiguity regarding the linear component, if any, of $\mathbf{f}$, $\tilde{f}$, g, $\tilde{g}$. Reformulate the velocity-based nonlinear system, (14), as in figure 1; that is, as the nonlinear system

$$\begin{aligned} \dot{\rho} &= \nabla_{r_{aug}}\rho\, \dot{r}_{aug}, \dot{\omega} = \mathbf{A}\omega + \left(\begin{bmatrix}\mathbf{B} & 0\end{bmatrix} + \nabla\mathbf{f}(\rho)\nabla_{r_{aug}}\rho\right)\dot{r}_{aug} \\ \dot{\eta} &= \mathbf{C}\omega + \left(\begin{bmatrix}\mathbf{D} & 0\end{bmatrix} + \nabla\mathbf{g}(\rho)\nabla_{r_{aug}}\rho\right)\dot{r}_{aug}, \dot{\eta}_\rho = \mathbf{M}\,\omega + \begin{bmatrix}\mathbf{N} & 0\end{bmatrix}\dot{r}_{aug} \end{aligned} \tag{16}$$

with augmented input $\dot{r}_{aug} = \begin{bmatrix}\dot{r} \\ \dot{\rho}\end{bmatrix}$, enclosed within a unity feedback loop by setting the input, $\dot{\rho}$, equal to the output, $\dot{\eta}_\rho$ (where $\mathbf{M} = \nabla_x\rho$, $\mathbf{N} = \nabla_r\rho$ and $\nabla_{r_{aug}}\rho = \begin{bmatrix}0 & \mathbf{I}\end{bmatrix}$). Clearly, the nonlinear system, (16), is also in velocity-based form but with the scheduling variable, ρ, is now an input. Similarly, reformulate (15) as

$$\begin{aligned} \dot{\tilde{\rho}} &= \nabla_{\tilde{r}_{aug}}\tilde{\rho}\,\dot{\tilde{r}}_{aug}, \dot{\tilde{\omega}} = \tilde{\mathbf{A}}\tilde{\omega} + \left(\begin{bmatrix}\tilde{\mathbf{B}} & 0\end{bmatrix} + \nabla\tilde{\mathbf{f}}(\tilde{\rho})\nabla_{\tilde{r}_{aug}}\tilde{\rho}\right)\dot{\tilde{r}}_{aug} \\ \dot{\tilde{\eta}} &= \tilde{\mathbf{C}}\tilde{\omega} + \left(\begin{bmatrix}\tilde{\mathbf{D}} & 0\end{bmatrix} + \nabla\tilde{\mathbf{g}}(\tilde{\rho})\nabla_{\tilde{r}_{aug}}\tilde{\rho}\right)\dot{\tilde{r}}_{aug}, \dot{\tilde{\eta}}_\rho = \tilde{\mathbf{M}}\,\tilde{\omega} + \begin{bmatrix}\tilde{\mathbf{N}} & 0\end{bmatrix}\dot{\tilde{r}}_{aug} \end{aligned} \tag{17}$$

with augmented input $\dot{\tilde{r}}_{aug} = \begin{bmatrix} \dot{r} \\ \dot{\tilde{\rho}} \end{bmatrix}$, enclosed within a unity feedback loop by setting the input, $\dot{\tilde{\rho}}$, equal to the output, $\dot{\tilde{\eta}}_\rho$ (where $\tilde{M} = \nabla_{\tilde{x}}\tilde{\rho}$, $\tilde{N} = \nabla_r \tilde{\rho}$ and $\nabla_{\tilde{r}_{aug}}\tilde{\rho} = [0 \quad I]$). The corresponding velocity-based linearisation families are obtained, respectively, by "freezing" (16) and (17).

Assume that corresponding members of the velocity-based linearisation families (i.e. for which $\rho=\rho_1$, $\tilde{\rho}=\rho_1$) have, respectively, the same transfer function from $\dot{r}$ to $\begin{bmatrix} \dot{\eta} \\ \dot{\eta}_\rho \end{bmatrix}$ and from $\dot{r}$ to $\begin{bmatrix} \dot{\tilde{\eta}} \\ \dot{\tilde{\eta}}_\rho \end{bmatrix}$. It follows immediately from standard linear theory that

$$\begin{aligned} \tilde{A} &= T(\rho_1)AT^{-1}(\rho_1), \quad \left([\tilde{B} \quad 0] + \nabla\tilde{f}(\tilde{\rho})\nabla_{\tilde{r}_{aug}}\tilde{\rho}\right) = T(\rho_1)\left([B \quad 0] + \nabla f(\rho)\nabla_{r_{aug}}\rho\right) \\ \tilde{C} &= CT^{-1}(\rho_1), \quad \left([\tilde{D} \quad 0] + \nabla\tilde{g}(\tilde{\rho})\nabla_{\tilde{r}_{aug}}\tilde{\rho}\right) = \left([D \quad 0] + \nabla g(\rho)\nabla_{r_{aug}}\rho\right) \\ \tilde{M} &= MT^{-1}(\rho_1), \quad \tilde{N} = N \end{aligned} \tag{18}$$

where $T(\rho_1)$ is a non-singular linear state transformation (which may be different for each member of a linear family). Without loss of generality, let $T(0)$ be the identity matrix; then owing to the minimality condition 3, it follows that (18) reduces at the origin to $\tilde{A}=A$, $\tilde{B}=B$, $\tilde{C}=C$, $\tilde{D}=D$, $\tilde{M}=M$, $\tilde{N}=N$. Hence, $T(\rho_1)AT^{-1}(\rho_1) = A, CT^{-1}(\rho_1) = C$ and so

$$\left[C \quad CA \quad \cdots \quad CA^n\right]^T T^{-1}(\rho_1) = \left[C \quad CA \quad \cdots \quad CA^n\right]^T \qquad \forall\rho_1 \tag{19}$$

From the observability of (A, C), the matrix $\left[C \quad CA \quad \cdots \quad CA^n\right]^T$ is full rank and it follows from (19) that $T(\rho_1)$ must be constant; that is, $T(\rho_1)=T(0)=I$. Consequently, under the foregoing conditions the nonlinear systems (14) and (15) (and so (12) and (13)) must be identical.

Summary

The reconstruction of a nonlinear dynamic system from a suitable collection of identified linear systems is considered. It is shown that knowledge only of the transfer functions of the linearisations of the nonlinear system is insufficient to permit such reconstruction and sufficient conditions based on augmented transfer function knowledge are derived.

Acknowledgement

D.J.Leith gratefully acknowledges the support provided by the Royal Society for the work presented.

References

LEITH, D.J., LEITHEAD, W.E., 1998a, Gain-Scheduled & Nonlinear & Systems: Dynamic Analysis by Velocity-Based Linearisation Families. *Int. J. Control*, 70, pp289-317; 1998b, Gain-Scheduled Controller Design: An Analytic Framework Directly Incorporating Non-Equilibrium Plant Dynamics. *ibid*, 70, pp249-269.

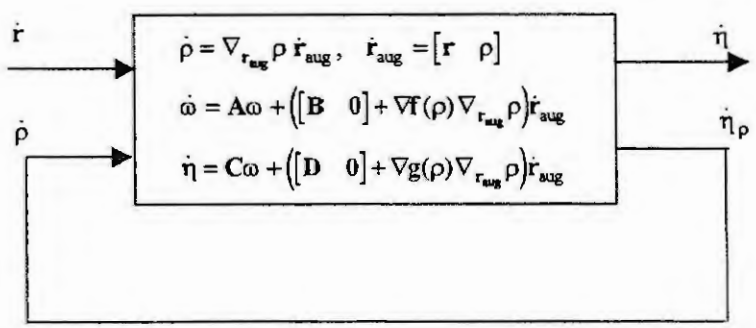


Figure 1 Reformulation to provide additional information from identified transfer function.

TURNING VECTOR PARTIAL DIFFERENTIAL EQUATIONS INTO MULTIDIMENSIONAL TRANSFER FUNCTION MODELS

L. Trautmann and R. Rabenstein
Telecommunications Laboratory, University of Erlangen-Nuremberg
D-91058 Erlangen, Cauerstr. 7, Germany
Email: traut,rabe@lnt.de

Abstract. Transfer function models for the description of physical systems have recently been introduced. They provide an alternative to the conventional representation by partial differential equations (PDE) and are suitable for computer implementation. This paper presents the transfer function models for vector PDEs. They arise from the physical analysis of multidimensional systems in terms of potential and flux quantities. Expressing the resulting coupled PDEs in vector form facilitates the direct formulation of boundary and interface conditions in their physical context. It is shown how a carefully constructed transformation for the space variable leads to transfer function models for vector PDEs. They are the starting point for the derivation of discrete models by standard methods for one-dimensional systems. The presented functional transformation approach is suitable for a number of technical applications, like electromagnetics, optics, acoustics and heat and mass transfer.

1 Introduction

The conventional tools for the description of time and space dependent physical phenomena are partial differential equations (PDEs). Their derivation usually involves two dependent physical variables: a potential and a flux quantity. The relationship between these quantities can be established from the first principles of physics. This results in a pair of coupled PDEs. They may be written in a compact form by combining the potential and flux quantities into one vector of unknowns and by arranging the differential operators in matrix notation. This form is called a vector PDE. Vector PDEs are not suitable for direct computer implementation since they contain spatial and temporal derivatives.

From one-dimensional systems given by ODEs it is well known that a description with transfer function models can be achieved by applying the Laplace transformation. These transfer functions are an established model in linear systems theory, electrical network theory, and control theory. They are not only indispensible for theoretical considerations, they also allow to derive effective algorithms.

Multidimensional (MD) systems given by scalar PDEs can also be described with MD transfer function models by applying suitable spatial transformations [3]. The transformation kernels can be calculated from the eigenvalues of the self-adjoint spatial differential operators of the scalar PDEs. The transformation kernels can also be used for inverse transformation since they are orthogonal.

The extension to vector PDEs with two or more output variables leads to non-self-adjoint spatial operators. Consequently the transformation kernels are no longer orthogonal. Thus the kernels of the inverse transformation must be calculated from the eigenvalues of the adjoint eigenvalue problem of the foreward transformation. Then the transformation kernel of the foreward and the inverse transformation form a biorthogonal system.

The advantage of using MD transfer function models derived from vector PDEs instead of scalar PDEs is that both potential and flux quantities occur as variables of the model. Thus interfaces between seperate spatial domains with different physical properties and thus different eigenvalues can be modeled more simply than with transfer function models derived from scalar PDEs.

The resulting MD transfer function models are an excellent starting point for the development of discrete-variable representations of physical systems. They offer the following advantages: suitability for computer implementation by iteration free algorithms, stability in the sense of systems theory, and high numerical accuracy without prohibitively small step sizes [2].

2 Problem Description

We consider an initial-boundary-value problem with time t and one space dimension x as independent variables. The potential $u(x,t)$ and the flux $i(x,t)$ are combined into the vector $\mathbf{y}(x,t) = [u(x,t) \quad i(x,t)]^T$.

The system of coupled PDEs for potential and flux is of the generic homogeneous form

$$\mathbf{C}D_t\mathrm{y}(x,t) + \mathbf{L}\mathrm{y}(x,t) = \mathbf{0} \quad \text{with} \quad \mathbf{L} = \mathbf{A} + \mathbf{B}D_x. \tag{1}$$

D_t denotes derivation with respect to time. $\mathbf{C}$ is a mass or capacitance matrix. $\mathbf{L}$ is a matrix operator containing loss terms in $\mathbf{A}$ and spatial derivatives in $\mathbf{B}D_x$. The initial conditions are given by

$$\mathrm{y}(x,0) = \mathrm{y}_i(x), \quad t = 0. \tag{2}$$

At the endpoints of the interval $x_0 < x < x_1$, we set boundary conditions by a suitable combination of potential and flux, expressed by the boundary operator $\mathbf{f}_b$ (H denotes matrix conjugate transpose).

$$\mathbf{f}_{b,n}^H \mathrm{y}(x_n,t) = \phi_n(t), \quad n = 0,1. \tag{3}$$

3 Transfer Function Model

The derivation of a transfer function model for vector PDEs follows the same steps as for the scalar case [2, 3]:

1. Apply the Laplace transformation with respect to time. This removes the time derivatives and turns the initial-boundary-value problem into a boundary value problem for the space variable.
2. Construct a suitable transformation for the space variable which removes the spatial derivatives and turns the boundary value problem into an algebraic equation.
3. To obtain a MD transfer function, solve the algebraic equation for the transform of the solution of the PDE.

From this MD transfer function a discrete model in the form of a MD difference equation can be derived which is suitable for computer implementation.

3.1 Temporal transformation

To remove the temporal derivatives of (1) and to include the initial values into the algebraic equation we apply the Laplace transformation. We obtain from (1-3) with the temporal frequency variable s

$$s\mathbf{C}\mathrm{Y}(x,s) + \mathbf{L}\mathrm{Y}(x,s) = \mathbf{C}\mathrm{y}_i(x) \tag{4}$$

$$\mathbf{f}_{b,n}^H \mathrm{Y}(x_n,s) = \Phi_n(s), \quad n = 0,1 \quad . \tag{5}$$

3.2 Spatial transformation

The crucial point in the derivation of the transfer function model is the transformation for the space variable. Its construction will be discussed in detail.

3.2.1 Transformation formulation

Since we want to remove the spatial derivatives from (4) and we want to include the boundary conditions into the resulting algebraic equation we formulate a spatial transformation which is similar to the Laplace transformation for the time variable. But due to the different integration limits we obtain different transformation kernels $\bar{\mathbf{K}}^H(x,\tilde{\beta}_\mu)$ that are not known a priori. This transformation formulation can be interpreted as a generalized Fourier expansion

$$\mathcal{T}\{\mathrm{Y}(x)\} = \bar{Y}(\tilde{\beta}_\mu) = \int_{x_0}^{x_1} \bar{\mathbf{K}}^H(x,\tilde{\beta}_\mu)\mathbf{C}\mathrm{Y}(x)dx\,. \tag{6}$$

(6) transforms the vector $\mathrm{Y}(x)$ into the scalar $\bar{Y}(\tilde{\beta}_\mu)$ depending on the spatial frequency variable $\tilde{\beta}_\mu$ similar to the temporal frequency variable s. From (4) can be seen that the weighting matrix $\mathbf{C}$ cannot be eliminated and must be integrated into the transformation formulation.

3.2.2 Differentiation theorem

With (6) the first summand on the left hand side of (4) and the initial conditions $\mathbf{y}_i(x)$ can be directly transformed into the spatial frequency domain. For the operator $\mathbf{L}$ of the second summand in (4) containing spatial derivatives we have to formulate a differentiation theorem. It should have the same properties as the differentiation theorem of the Laplace transformation. We can write ($*$ denotes conjugation)

$$\int_{x_0}^{x_1} \bar{\mathbf{K}}^H(x,\tilde{\beta}_\mu)\mathbf{L}\mathbf{Y}(x)dx = \int_{x_0}^{x_1} \left[\tilde{\mathbf{L}}\bar{\mathbf{K}}\right]^H \mathbf{Y}dx + \left[\bar{\mathbf{K}}^H\mathbf{B}\mathbf{Y}\right]_{x_0}^{x_1} \tag{7}$$

$$\left[\bar{\mathbf{K}}^H\mathbf{B}\mathbf{Y}\right]_{x_0}^{x_1} = \left[\tilde{\mathbf{f}}_{b,1}^H\bar{\mathbf{K}}^*\cdot\mathbf{g}_{b,1}^H\mathbf{Y} - \tilde{\mathbf{g}}_{b,1}^H\bar{\mathbf{K}}^*\cdot\mathbf{f}_{b,1}^H\mathbf{Y}\right]_{x=x_1} - \left[\tilde{\mathbf{f}}_{b,0}^H\bar{\mathbf{K}}^*\cdot\mathbf{g}_{b,0}^H\mathbf{Y} - \tilde{\mathbf{g}}_{b,0}^H\bar{\mathbf{K}}^*\cdot\mathbf{f}_{b,0}^H\mathbf{Y}\right]_{x=x_0} \tag{8}$$

(7) is an extended Green's formula. In this context the operator $\tilde{\mathbf{L}} = \mathbf{A}^H - \mathbf{B}^H D_x$ is called the adjoint operator of $\mathbf{L} = \mathbf{A} + \mathbf{B}D_x$ [1]. It has several properties that are explained later. In (8) the boundary conditions are divided into the given parts $\mathbf{f}_{b,n}^H$ and the unknown parts $\mathbf{g}_{b,n}^H$ of the output variable $\mathbf{Y}$.

3.2.3 Determination of the transformation kernel

To obtain an algebraic equation in the spatial frequency domain we have to express the integral term of (7) as a scalar multiplication of the spatial frequency variable $\tilde{\beta}_\mu$ with the output variable $\bar{Y}(\tilde{\beta}_\mu)$. This leads to the first condition for the determination of the transformation kernel:

$$\int_{x_0}^{x_1} \left[\tilde{\mathbf{L}}\bar{\mathbf{K}}\right]^H \mathbf{Y}dx \;=\; \tilde{\beta}_\mu\bar{Y}(\tilde{\beta}_\mu) = \int_{x_0}^{x_1} \tilde{\beta}_\mu\bar{\mathbf{K}}^H\mathbf{C}\mathbf{Y}dx \tag{9}$$

Since the integrals in (9) should be equal for every function $\mathbf{Y}(x)$ we obtain

$$\left[\tilde{\mathbf{L}}\bar{\mathbf{K}}\right]^H = \tilde{\beta}_\mu\bar{\mathbf{K}}^H\mathbf{C}\,. \tag{10}$$

If $\mathbf{B}$ has full rank we derive from (1,10) (T denotes matrix transpose)

$$D_x\bar{\mathbf{K}}^* = \bar{\mathbf{Q}}\bar{\mathbf{K}}^* \quad \text{with} \quad \bar{\mathbf{Q}} = \left(\mathbf{B}^T\right)^{-1}\left(\mathbf{A}^T - \tilde{\beta}_\mu\mathbf{C}^T\right). \tag{11}$$

The solution of this matrix eigenvalue yields the eigenvalues and eigenvectors of the matrix $\bar{\mathbf{Q}}$.

The second and third equation for the transformation kernel are derived from (8). Since the boundary conditions $\mathbf{g}_{b,n}^H\mathbf{Y}$ at $x = x_n$, $n = 0,1$ are not known the summand with these terms should vanish. That can be obtained by demanding

$$\tilde{\mathbf{f}}_{b,n}^H\bar{\mathbf{K}}^*(x_n,\tilde{\beta}_\mu) = 0, \qquad n = 0,1\,. \tag{12}$$

(10,12) is called a generalization of the Sturm-Liouville type problem. Therefore the spatial transformation (6) is called a Sturm-Liouville transformation. Solving this Sturm-Liouville type problem we obtain the transformation kernel $\bar{\mathbf{K}}$ and the spatial frequency variable $\tilde{\beta}_\mu$. From the mathematical point of view the transformation kernel represents the eigenfunctions and the spatial frequency variable represent the eigenvalues of the operator $\tilde{\mathbf{L}}$ with respect to the weighting matrix $\mathbf{C}$. Since $\tilde{\mathbf{L}}$ is a compact operator we only obtain discrete eigenvalues $\tilde{\beta}_\mu$.

3.3 Transfer function model

With the temporal transformation (4,5) and the spatial transformation (6,9) we can now turn the PDE with temporal and spatial derivatives (1) and its initial (2) and boundary conditions (3) into one multidimensional algebraic equation in the temporal and spatial frequency domain

$$\bar{Y}(\tilde{\beta}_\mu,s) = \frac{\bar{\Phi}_{b,1}(\tilde{\beta}_\mu,s) - \bar{\Phi}_{b,0}(\tilde{\beta}_\mu,s) + \bar{\mathrm{y}}_i(\tilde{\beta}_\mu)}{s+\tilde{\beta}_\mu} \quad \text{with} \quad \bar{\Phi}_{b,n}(\tilde{\beta}_\mu,s) = \Phi_n(s)\cdot\tilde{\mathbf{g}}_b^H\bar{\mathbf{K}}^*(x_n,\tilde{\beta}_\mu), \quad n = 0,1\,. \tag{13}$$

The boundary conditions are also described in the spatial frequency domain. (13) is the transfer function model for the MD problem, initially described by a PDE with initial and boundary conditions. The initial $\bar{y}_i(\tilde{\beta}_\mu)$ and boundary conditions $\bar{\Phi}_{b,n}(\tilde{\beta}_\mu, s)$ are now included as inputs in the transform domain in the numerator of (13). The denominator is the MD transfer function that filters the input signals to the output variable $\bar{Y}(\tilde{\beta}_\mu, s)$.

3.4 Inverse transformation

Since the inverse Laplace transformation is well known, only the inverse spatial transformation is discussed in detail here. It is formulated as a generalized Fourier series since the frequency variable $\tilde{\beta}_\mu$ is discrete. N_μ is a norm factor.

$$\mathbf{Y}(x,s) = \mathcal{T}^{-1}\{\bar{Y}(\tilde{\beta}_\mu, s)\} = \sum_{\mu=-\infty}^{\infty} \frac{1}{N_\mu} \bar{Y}(\tilde{\beta}_\mu, s)\mathbf{K}(x, \tilde{\beta}_\mu)\,. \tag{14}$$

Since the elements of $\tilde{\mathbf{K}}(x, \tilde{\beta}_\mu)$ are not necessarily orthogonal, we have to find a transformation kernel $\mathrm{K}(x, \beta_\mu)$ for the inverse transformation that is biorthogonal to $\tilde{\mathbf{K}}(x, \tilde{\beta}_\mu)$. $\mathbf{K}$ results from the adjoint eigenvalue problem of (10) and (12):

$$\mathbf{L}\mathbf{K}(x, \beta_\mu) = \beta_\mu \mathbf{C}\mathbf{K}(x, \beta_\mu)\,, \tag{15}$$

$$\mathbf{f}_{b,n}^H \mathbf{K}(x_n, \beta_\mu) = 0, \quad n = 0, 1 \tag{16}$$

K and $\tilde{\mathbf{K}}$ are biorthogonal functions with respect to the weighting matrix $\mathbf{C}$. This can be shown by applying Green's formula. Replacing $\mathbf{Y}$ in (7) with $\mathbf{K}$ we obtain

$$\int_{x_0}^{x_1} \tilde{\mathbf{K}}^H(\tilde{\beta}_\mu)\mathbf{L}\mathbf{K}(\beta_\nu)dx - \int_{x_0}^{x_1} \left[\tilde{\mathbf{L}}\tilde{\mathbf{K}}(\tilde{\beta}_\mu)\right]^H \mathbf{K}(\beta_\nu)dx = \left[\tilde{\mathbf{f}}_b^H \tilde{\mathbf{K}}^* \cdot \mathbf{g}_b^H \mathbf{K} - \tilde{\mathbf{g}}_b^H \tilde{\mathbf{K}}^* \cdot \mathbf{f}_b^H \mathbf{K}\right]_{x_0}^{x_1}. \tag{17}$$

The right-hand side of (17) must be zero with respect to (12) and (16) and the left-hand side of (17) can be evaluated to (18) with respect to (10) and (15).

$$\int_{x_0}^{x_1} \tilde{\mathbf{K}}^H(\tilde{\beta}_\mu)\mathbf{L}\mathbf{K}(\beta_\nu)dx - \int_{x_0}^{x_1} \left[\tilde{\mathbf{L}}\tilde{\mathbf{K}}(\tilde{\beta}_\mu)\right]^H \mathbf{K}(\beta_\nu)dx = \left(\beta_\nu - \tilde{\beta}_\mu\right) \int_{x_0}^{x_1} \tilde{\mathbf{K}}^H(\tilde{\beta}_\mu)\mathbf{C}\mathbf{K}(\beta_\nu)dx \tag{18}$$

Since the first factor of the right hand side of (18) is only zero for $\tilde{\beta}_\mu = \beta_\nu$, the second factor must be zero for $\tilde{\beta}_\mu \neq \beta_\nu$. Since $\tilde{\mathbf{L}}$ is the adjoint operator of $\mathbf{L}$ it can be shown that their eigenvalues are related by $\tilde{\beta}_\mu = \beta_\mu^*$. Since every eigenvalue $\tilde{\beta}_\mu$ has one conjugate counterpart, $\mathbf{K}$ and $\tilde{\mathbf{K}}$ are biorthogonal functions with respect to the weighting matrix $\mathbf{C}$. N_μ can be evaluated by inserting (14) into (6).

4 Conclusions

The transformation of scalar PDEs into MD transfer function models was extended in this paper to vector PDEs. We have shown that this physically based approach leads to a transformation formulation with non-self-adjoint operators. It resulted in a set of biorthogonal transformation kernels. Although presented here in only one spatial dimension it can be extended to two or more dimensions without loss of generality. This method is used to obtain effective algorithms for real-time implementations of physical problems usually described by PDEs, like heat- and mass-transfer, optics and acoustics.

References

[1] C. Lanzcos. *Linear Differential Operators.* D. Van Nostrand Company, London, 1961.

[2] R. Rabenstein. Discrete simulation models for multidimensional systems based on functional transformations. In J.G. McWhirter and I.K. Proudler, editors, *Mathematics in Signal Processing IV*, pages 335–347. Oxford University Press, 1998.

[3] R. Rabenstein. Transfer function models for multidimensional systems with bounded spatial domains. *Mathematical and Computer Modelling of Dynamical Systems*, 5(3):259–278, 1999.

Two approaches for state space realization of NARMA models: bridging the gap

Ü. Kotta[1] and N. Sadegh[2]

[1]Institute of Cybernetics, Tallinn TU,
Akadeemia tee 21, Tallinn, 12618, Estonia, Email: kotta@ioc.ee
[2]The G.W. Woodruff School of Mechanical Engineering,
Georgia Institute of Technology, Atlanta, Georgia 30332-0405, USA

Abstract

This paper presents the necessary and sufficient conditions for observable realization of a general class of nonlinear input–output maps. In particular, it proves the equivalence of the two seemingly different existing approaches in the literature. The paper also provides a general class of NARMA input–output models that are guaranteed to have an observable realization. It is shown that this class covers several important subclasses of existing NARMA models.

1 Introduction

The state space realization of single–input single–output nonlinear input–output (i/o) difference equations has been the subject of two recent papers [1, 2]. In these papers two completely different approaches were provided to solve the problem. In [1], the intrinsic and coordinate-free generic necessary and sufficient realizability conditions were formulated in terms of integrability of certain subspaces of one-forms associated with the i/o model. One of the distinctive characteristics which makes this algebraic approach interesting and useful is its inherent simplicity and transparency. For characterizing the realizability conditions as well as for constructing the state coordinates, a single tool, based on elementary time shifting of a function (and an one-form), namely the notion of relative degree, provides the key. Despite the inherent simplicity of the approach, in order to find the state coordinates, one has to integrate the integrable one-forms of a certain subspace. Though it is easy to check the integrability property with the help of the Frobenius theorem, it can be extremely difficult to integrate the required one-forms to find the state coordinates, especially for complicated models.

On the other hand, the local necessary and sufficient realizability conditions in [2] are formulated directly in terms of the partial derivatives of the i/o map of the NARMA model. Moreover, the state coordinates can be constructed based on the above i/o map. The most complicated task in finding the state coordinates is to solve a nonlinear algebraic equation, and not a set of nonlinear differential equations as in [1].

The main contributions of this paper is to prove that the two seemingly different approaches are actually equivalent and yield the same results. The algorithm in [2] can be understood as the method to compute the basis for a subspace of one-forms. Particularly, this equivalence allows to extend the results of [1] by providing an (almost) explicit choice of the state coordinates as in [2] without the need to integrate the one-forms and to solve the set of nonlinear differential equations. This is especially important for implementation of the realization algorithms via the symbolic computation packages such as *Mathematica* or *Maple*. Solving the set of nonlinear differential equations and integrating the one-forms are reported to be the most complicated tasks for computer algebra implementation; all the other operations related to the realization procedure, can be implemented rather straightforwardly [3].

Moreover, the equivalence of the two approaches provides a more general point of view for the results in [2]. In particular, it allows extension of the local results of [2] to global using the more sophisticated

algebraic framework as in [1].

The second purpose of the paper is to provide a more general class of realizable NARMA models as the one suggested in [2].

2 Equivalence of the necessary and sufficient realizability conditions

Consider a nonlinear system Σ described by

$$y(t+n) = \varphi(y(t), \ldots, y(t+n-1), u(t), \ldots, u(t+n-1)) \tag{1}$$

where $u \in \mathcal{U} \subset \mathbb{R}$ is the input variable, $y \in \mathcal{Y} \subset \mathbb{R}$ is the scalar output variable and φ is a real analytic function defined on $\mathcal{Y}^n \times \mathcal{U}^n$. Assume that either $\partial\varphi(\cdot)/\partial y(t)$ or $\partial\varphi(\cdot)/\partial u(t)$ is different from zero (Assumption 1). In the realization problem we are looking for transforming an i/o equation (1) into the classical state-space form

$$\begin{aligned} x(t+1) &= f(x(t), u(t)) \\ y(t) &= h(x(t)). \end{aligned} \tag{2}$$

We will associate with the system Σ an extended state-space system Σ_e with input $v(t) = u(t+s+1)$ and state $z(t) = [y(t), \ldots, y(t+n-1), u(t), \ldots, u(t+s)]^T$ defined as

$$z(t+1) = f_e(z(t), v(t)) \tag{3}$$

where $f_e(\cdot) = [z_2, \ldots, z_n, \phi(z), z_{n+2}, \ldots, z_{n+s+1}, v]^T$. The system (3) will play a key role in the realizability conditions and the realization procedure of [1].

Let $\mathcal{K}$ denote the field of meromorphic functions in a finite number of variables $\{z(0), v(t), t \geq 0\}$. The forward-shift operator $\delta : \mathcal{K} \to \mathcal{K}$ is defined by

$$\delta\zeta(z(t), v(t)) = \zeta(f_e(z(t), v(t)), v(t+1))$$

The pair $(\mathcal{K}, \delta)$ is a difference field. and up to an isomorphism, there exists a unique difference field $(\mathcal{K}^*, \delta^*)$ such that $\mathcal{K} \subset \mathcal{K}^*, \delta^* : \mathcal{K}^* \to \mathcal{K}^*$ is an automorphism and the restriction of δ^* to $\mathcal{K}$ equals δ. By abuse of notation, hereinafter we assume that the inversive closure $(\mathcal{K}^*, \delta^*)$ is given and use the same symbol to denote the difference field $(\mathcal{K}, \delta)$ and its inversive closure.

We first define generic observability for system (2):

Definition 2.1 *We call system (2) locally generically observable if*

$$\operatorname{rank}_{\mathcal{K}} \frac{\partial(y(t), y(t+1), \ldots, y(t+n-1))}{\partial x(t)} = n. \tag{4}$$

Over the field $\mathcal{K}$ one can define a difference vector space $\mathcal{E} := \operatorname{span}_{\mathcal{K}}\{\mathrm{d}\varphi \mid \varphi \in \mathcal{K}\}$. The operator δ induces a forward-shift operator $\Delta : \mathcal{E} \to \mathcal{E}$ by

$$\sum_i a_i \mathrm{d}\varphi_i \mapsto \sum_i (\delta a_i)\mathrm{d}(\delta\varphi_i),\ a_i, \varphi_i \in \mathcal{K}.$$

The relative degree r of a one-form $\omega \in \mathcal{E}$ is defined to be the least integer such that $\Delta^r\omega \notin \operatorname{span}_{\mathcal{K}}\{\mathrm{d}z\}$. If such an integer does not exist, we set $r = \infty$. A sequence of subspaces $\{\mathcal{H}_k\}$ of $\mathcal{E}$ is defined by

$$\begin{aligned} \mathcal{H}_1 &= \operatorname{span}_{\mathcal{K}}\{\mathrm{d}z(0)\} = \operatorname{span}_{\mathcal{K}}\{\mathrm{d}y(0), \ldots, \mathrm{d}y(n-1), \mathrm{d}u(0), \ldots, \mathrm{d}u(s)\} \\ \mathcal{H}_{k+1} &= \operatorname{span}_{\mathcal{K}}\{\omega \in \mathcal{H}_k \mid \Delta\omega \in \mathcal{H}_k\},\ k \geq 1 \end{aligned} \tag{5}$$

It is clear that sequence (5) is decreasing. The subspaces are invariant under the (extended) state space diffeomorphism.

Theorem 2.2 *The nonlinear system described by the input-output difference equation (1) has a generically observable state space realization iff for $1 \leq k \leq n+1$ the subspaces $\mathcal{H}_k$ defined by (5) are completely integrable. Moreover, the state coordinates can be obtained by integrating the one-forms in $\mathcal{H}_{n+1}$.*

Next, we will recall the realizability conditions, given in [2]. For that, we define the blocks of input and output by

$$\begin{aligned} \mathbf{u}(t) &= (u(t), \ldots, u(t+n-1)), \\ \mathbf{y}(t) &= (y(t), \ldots, y(t+n-1)) \end{aligned}$$

Evaluating $y(t), y(t+1), \ldots, y(t+n-1)$ recursively in terms of $\mathbf{y}(t-n), \mathbf{u}(t-n)$ and $\mathbf{u}(t)$ we obtain the block i/o map

$$\mathbf{y}(t) = \Phi(\mathbf{y}(t-n), \mathbf{u}(t-n), \mathbf{u}(t))$$

where $\Phi = (\varphi^1, \varphi^2 \ldots, \varphi^n)^T$, $\varphi^1(\mathbf{y}, \mathbf{u}, \mathbf{v}) = \varphi(y_1, \ldots, y_n, u_1, \ldots, u_n)$, and

$$\varphi^i(\mathbf{y}, \mathbf{u}, \mathbf{v}) = \varphi(y_i, \ldots, y_n, \varphi^1(\mathbf{y}, \mathbf{u}, \mathbf{v}), \ldots, \varphi^{i-1}(\mathbf{y}, \mathbf{u}, \mathbf{v}), u_i, \ldots, u_n, v_1, \ldots, v_{i-1})$$

for $i = 2, \ldots n$. Note that in terms of the forward–shift operator δ, $\varphi^i(\mathbf{y}, \mathbf{u}, \mathbf{v}) = \delta^{i-1}\varphi(y_1, \ldots, y_m, u_1, \ldots, u_m)$. For $\mathbf{x} \in \mathbb{R}^n$, denoting $D_{\mathbf{x}} := \left[\begin{array}{ccc} \frac{\partial}{\partial x_1} & \cdots & \frac{\partial}{\partial x_n} \end{array} \right]$, it can be easily seen that Assumption 1 implies that either $D_{\mathbf{y}}\Phi(\mathbf{y}, \mathbf{u}, \mathbf{v})$ or $D_{\mathbf{u}}\Phi(\mathbf{y}, \mathbf{u}, \mathbf{v})$ is invertible. To accommodate, these two cases we define the modified block i/o map $\hat{\Phi}(\mathbf{y}, \mathbf{u}, \mathbf{v}) := \Phi(\mathbf{y}, \mathbf{u}, \mathbf{v})$ if $\partial\varphi(\cdot)/\partial y(t)$ is nonzero and $\hat{\Phi}(\mathbf{u}, \mathbf{y}, \mathbf{v}) := \Phi(\mathbf{y}, \mathbf{u}, \mathbf{v})$ otherwise. Using the implicit function theorem, it can be seen that the map $\hat{\Phi}$ $(\mathbf{y}, \mathbf{u}, \mathbf{v})$ is a local diffeomorphism with respect to $\mathbf{y}$. That is, there exists a smooth local function $\hat{\Phi}_y^{-1}$ such that $\mathbf{y} = \hat{\Phi}_y^{-1}(\mathbf{z}, \mathbf{u}, \mathbf{v}) \Longrightarrow \mathbf{z} = \hat{\Phi}(\mathbf{y}, \mathbf{u}, \mathbf{v})$. For future reference we need $D_{\mathbf{u}}\hat{\Phi}_y^{-1}(\mathbf{z}, \mathbf{u}, \mathbf{v})$. Taking the partial derivative of $\mathbf{z} = \hat{\Phi}(\hat{\Phi}_y^{-1}(\mathbf{z}, \mathbf{u}, \mathbf{v}), \mathbf{u}, \mathbf{v})$ with respect to $\mathbf{u}$ it follows that

$$D_{\mathbf{u}}\hat{\Phi}_y^{-1}(\mathbf{z}, \mathbf{u}, \mathbf{v}) = -\left(D_{\mathbf{y}}\hat{\Phi}(\mathbf{y}, \mathbf{u}, \mathbf{v})\right)^{-1} D_{\mathbf{u}}\hat{\Phi}(\mathbf{y}, \mathbf{u}, \mathbf{v}) \tag{6}$$

We now present the main result of the paper, the proof of which shows explicitly the equivalence of the realizability conditions in [1] and a generalization of those in [2].

Theorem 2.3 *The nonlinear system described by the input-output difference equation (1) has a generically observable state space realization iff if $D_{\mathbf{y}}\hat{\Phi}(\mathbf{y}, \mathbf{u}, \mathbf{v})^{-1} D_{\mathbf{u}}\hat{\Phi}(\mathbf{y}, \mathbf{u}, \mathbf{v})$ is independent of the third variable $\mathbf{v}$. Moreover, the state vector may be defined by $x(t) = \hat{\Phi}(\mathbf{y}(t-n), \mathbf{u}(t-n), \mathbf{v})$ for any constant vector $\mathbf{v} \in \mathbb{R}^n$.*

Proof. We need to show that integrability of $\mathcal{H}_{n+1}$ is equivalent to the necessary and sufficient condition of Theorem 2.3.

Let $\omega = Ad\mathbf{y}(0) + Bd\mathbf{u}(0) \in \mathcal{H}_1$, with $A, B \in \mathcal{K}^{1\times n}$. Then $\Delta^n\omega = \delta^n Ad\mathbf{y}(n) + \delta^n Bd\mathbf{u}(n)$. Since $\mathbf{y}(n) = \Phi(\mathbf{y}(0), \mathbf{u}(0), \mathbf{u}(n))$,

$$d\mathbf{y}(n) = D_{\mathbf{y}(0)}\Phi(\mathbf{y}(0), \mathbf{u}(0), \mathbf{u}(n))d\mathbf{y}(0) + D_{\mathbf{u}(0)}\Phi(\mathbf{y}(0), \mathbf{u}(0), \mathbf{u}(n))d\mathbf{u}(0) + D_{\mathbf{u}(n)}\Phi(\mathbf{y}(0), \mathbf{u}(0), \mathbf{u}(n))d\mathbf{u}(n).$$

For $\Delta^n\omega \in \mathcal{H}_1$, one must have $\delta^n AD_{\mathbf{u}(n)}\Phi(\mathbf{y}(0), \mathbf{u}(0), \mathbf{u}(n)) + \delta^n B = 0$ which yields $B = -AD_{\mathbf{u}(0)}\Phi(\mathbf{y}(-n), \mathbf{u}(-n), \mathbf{u}(0))$. From the modified block i/o map, there exists a smooth local function $\hat{\Phi}_y^{-1}$ such that, $\hat{\mathbf{y}}(-n) = \hat{\Phi}_y^{-1}(\mathbf{y}(0), \hat{\mathbf{u}}(-n), \mathbf{u}(0))$ where $(\hat{\mathbf{u}}, \hat{\mathbf{y}}) = (\mathbf{u}, \mathbf{y})$ if $\partial\varphi(\cdot)/\partial y(t)$ is nonzero and $(\hat{\mathbf{u}}, \hat{\mathbf{y}}) = (\mathbf{y}, \mathbf{u})$ otherwise. Using this in the preceding equation, gives

$$B = -AD_{\mathbf{w}}\hat{\Phi}(\hat{\Phi}_y^{-1}(\mathbf{y}(0), \hat{\mathbf{u}}(-n), \mathbf{u}(0)), \hat{\mathbf{u}}(-n), \mathbf{w}) \mid_{\mathbf{w}=\mathbf{u}(0)}$$

Thus

$$\mathcal{H}_{n+1} = \mathrm{span}_{\mathcal{K}}\{d\mathbf{y}(0) - D_{\mathbf{w}}\hat{\Phi}(\hat{\Phi}_y^{-1}(\mathbf{y}(0), \hat{\mathbf{u}}(-n), \mathbf{u}(0)), \hat{\mathbf{u}}(-n), \mathbf{w}) \mid_{\mathbf{w}=\mathbf{u}(0)} d\mathbf{u}(0)\}.$$

It can be seen that $\mathcal{H}_{n+1}$ is integrable iff $D_{\mathbf{w}}\hat{\Phi}(\hat{\Phi}_y^{-1}(\mathbf{z}, \mathbf{u}, \mathbf{v}), \mathbf{u}, \mathbf{w}) \mid_{\mathbf{w}=\mathbf{v}}$ is independent of $\mathbf{u}$ or equivalently

$$\frac{\partial}{\partial\omega_i} D_{\mathbf{u}}\hat{\Phi}(\hat{\Phi}_y^{-1}(\mathbf{z}, \mathbf{u}, \mathbf{v}), \mathbf{u}, \mathbf{w}) \mid_{\mathbf{w}=\mathbf{v}} = 0$$

for $i = 1, \ldots, n$ where w_i is the i–th component of $\mathbf{w}$. Taking the derivative and using (6) yields

$$D_{\mathbf{u}}\hat{\Phi}(\hat{\Phi}_y^{-1}(\mathbf{z}, \mathbf{u}, \mathbf{v}), \mathbf{u}, \mathbf{w}) = D_{\mathbf{y}}\hat{\Phi}(\mathbf{y}, \mathbf{u}, \mathbf{w})(D_{\mathbf{u}}\hat{\Phi}_y^{-1}(\mathbf{z}, \mathbf{u}, \mathbf{v}) - D_{\mathbf{u}}\hat{\Phi}_y^{-1}(\mathbf{z}, \mathbf{u}, \mathbf{w}))$$

Thus

$$\frac{\partial}{\partial w_i} D_{\mathbf{u}} \hat{\Phi}(\hat{\Phi}_y^{-1}(\mathbf{z},\mathbf{u},\mathbf{v}),\mathbf{u},\mathbf{w}) \mid_{\mathbf{w}=\mathbf{v}} = -D_{\mathbf{y}} \hat{\Phi}(\mathbf{y},\mathbf{u},\mathbf{w}) \left(\frac{\partial}{\partial w_i} D_{\mathbf{u}} \hat{\Phi}_y^{-1}(\mathbf{z},\mathbf{u},\mathbf{w}) \mid_{\mathbf{w}=v} \right).$$

Since $D_{\mathbf{y}} \hat{\Phi}(\mathbf{y},\mathbf{u},\mathbf{w})$ is invertible, $\mathcal{H}_{n+1}$ is integrable iff $D_{\mathbf{u}} \hat{\Phi}_y^{-1}(\mathbf{z},\mathbf{u},\mathbf{v})$ is independent of $\mathbf{v}$ thereby proving the claim.

If $\mathcal{H}_{n+1}$ is integrable, then the same argument as the one in the sufficiency proof of Theorem 3.2 in [1] can be used to conclude integrability of all the $\mathcal{H}_j$'s.

Next we show that $dx(0) = d\hat{\Phi}(\hat{\Phi}_y^{-1}(\mathbf{y}(0),\mathbf{0},\mathbf{u}(0)),\mathbf{0},\mathbf{v}) \in \mathcal{H}_{n+1}$. Indeed,

$$dx = D_{\mathbf{y}} \hat{\Phi}(\mathbf{z},\mathbf{0},\mathbf{v})(D_{\mathbf{y}} \hat{\Phi}(\mathbf{z},\mathbf{0},\mathbf{u}))^{-1}(d\mathbf{y} - D_{\mathbf{u}} \hat{\Phi}(\mathbf{z},\mathbf{0},\mathbf{u})d\mathbf{u})$$

where $\mathbf{z} = \hat{\Phi}_y^{-1}(\mathbf{y},\mathbf{0},\mathbf{u})$. Thus $dx \in \mathcal{H}_{n+1}$.

The following corollary provides a sufficient condition for the realizability of the input-output difference equation (1) giving rise to a special subclass of NARMA models to be discussed.

Corollary 2.4 *The nonlinear system described by the input-output difference equation (1) has a generically observable state space realization if*

$$\alpha_{i,j}(\mathbf{y},\mathbf{u},\mathbf{v}) \;:\; = \delta^{i-1} D_{y_j} \varphi(y_1,\ldots,y_m,u_1,\ldots,u_m) \tag{7}$$

$$\beta_{i,j}(\mathbf{y},\mathbf{u},\mathbf{v}) \;:\; = \delta^{i-1} D_{u_j} \varphi(y_1,\ldots,y_m,u_1,\ldots,u_m) \tag{8}$$

are independent of $\mathbf{v}$ *for all* $i = 1,\ldots,n$, $j = 1,\ldots,n-i+1$.

Proof. It is shown in [2] that $D_{\mathbf{y}}\Phi = U^{-1}L_\alpha$ and $D_{\mathbf{u}}\Phi = U^{-1}L_\beta$ where $U(\mathbf{y},\mathbf{u},\mathbf{v})$ is an invertible lower triangular matrix, and $L_\alpha(\mathbf{y},\mathbf{u},\mathbf{v})$ and $L_\alpha(\mathbf{y},\mathbf{u},\mathbf{v})$ are upper triangular matrices involving only $\alpha_{i,j}(\mathbf{y},\mathbf{u},\mathbf{v})$ and $\beta_{i,j}(\mathbf{y},\mathbf{u},\mathbf{v})$, respectively. Thus the hypothesis implies that the upper triangular matrix $D_{\mathbf{y}}\hat{\Phi}(\mathbf{y},\mathbf{u},\mathbf{v})^{-1} D_{\mathbf{u}}\hat{\Phi}(\mathbf{y},\mathbf{u},\mathbf{v})$ is independent of $\mathbf{v}$. The proof of the Corollary now follows from Theorem 2.3 as a special case.

3 A special subclass of NARMA models

In many situations the NARMA model is obtained from experimental data using the identification procedures or neural networks. It is clear from Theorems 2.2 and 2.3 that an arbitrarily structured NARMA model does not necessarily have a state space realization. Using such a model is highly undesirable in further stability analysis and/or control design since practically all existing control theory for nonlinear systems bases on a state space description. Motivated by the above and relaying on the necessary and sufficient realizability conditions stated in Theorems 2.2 and 2.3, we now introduce a subclass of NARMA models, each of which is guaranteed to have a classical state space description.

The subclass provided by us is described by the following equations:

$$\begin{aligned} y(t+n) \;=\; & \varphi_{n-k}(y(t),\ldots,y(t+k),u(t)) + \varphi_{n-k-1}(y(t+1),\ldots+\ldots,y(t+k+1),u(t+1)) \\ & +\ldots+\varphi_1(y(t+n-k-1),\ldots,y(t+n-1),u(t+n-k-1)) \end{aligned} \tag{9}$$

for any $k = 0,1,\ldots,n-1$.

It can be easily checked that this NARMA satisfies the sufficient conditions of Corollary (2.4). Also, computing the subspace

$$\begin{aligned} \mathcal{H}_{n-k+1} \;=\; & \mathrm{sp}_{\mathcal{K}}\{dy(t),\ldots,dy(t+k), d[y(t+k+1)-\Phi_1(y(t+k),\ldots,y(t),u(t))], \\ & d[y(t+k+2)-\Phi_1(y(t+k+1),\ldots,y(t+1),u(t+1))-\Phi_2(y(t+k),\ldots,y(t),u(t)],\ldots \\ & d[y(t+n-1)-\Phi_1(y(t+n-2),y(t+n-3),\ldots,y(t+n-k-2), \\ & u(t+n-k-2))-\ldots-\Phi_{n-k-1}(y(t+k),\ldots,y(t),u(t))]\} \end{aligned}$$

shows that the NARMA model (9) is realizable in the classical state space form. The state coordinates can be obtained from Theorem (2.3) or by integrating the one-forms in $\mathcal{H}_{n-k+1}$. So, we choose

$$\begin{aligned}
x_1(t) &= y(t) \\
&\vdots \\
x_{k+1}(t) &= y(t+k) \\
x_{k+2}(t) &= y(t+k+1) - \Phi_1(y(t+k), \ldots, y(t), u(t)) \\
x_{k+3}(t) &= y(t+k+2) - \Phi_1(y+k+1), \ldots, y(t+1), u(t+1)) - \Phi_2(y(t+k), \ldots, y(t), u(t)) \\
x_n(t) &= y(t+n-1) - \Phi_1(y(t+n-2), \ldots, y(t+n-k-2), u(t+n-k-2)) \\
&\quad - \ldots \Phi_{n-k-1}(y(t+k), \ldots, y(t), u(t))
\end{aligned}$$

and obtain the state equations as follows

$$\begin{aligned}
x_1(t+1) &= x_2(t) \\
&\vdots \\
x_k(t+1) &= x_{k+1}(t) \\
x_{k+1}(t+1) &= x_{k+2}(t) + \Phi_1(x_{k+1}(t), \ldots, x_1(t), u(t)) \\
x_{k+2}(t+1) &= x_{k+3}(t) + \Phi_2(x_{k+1}(t), \ldots, x_1(t), u(t)) \\
&\vdots \\
x_{n-1}(t+1) &= x_n(t) + \Phi_{n-k-1}(x_{k+1}(t), \ldots, x_1(t), u(t)) \\
x_n(t+1) &= \Phi_{n-k}(x_{k+1}(t), \ldots, x_1(t), u(t))
\end{aligned} \tag{10}$$

Note that (9) covers several important subclasses of NARMA models. First, for $k = 1$ it reduces to the special subclass, presented in [2] and proved to be realizable in the Kalmanian form.

Second, for $k = 0$, it reduces to the subclass of nonlinear systems in the observer form.

Third, when studying the control problems for NARMA models, majority of the researchers typically assume the system to be in the controllability (Brunovsky) canonical form. This form corresponds to the case $k = n - 1$.

4 Conclusions

This paper explicitly proved the equivalence of the necessary and sufficient conditions in papers [1] and [2] for observable realization of a general class of nonlinear input–output maps. The paper also formulated a general class of realizable NARMA models, which covers several important subclasses of existing NARMA models.

5 Acknowledgment

This work was partially supported by the Estonian Science Foundation Grant Number 3137.

References

[1] Kotta, Ü., Liu, P. and Zinober, A. A state space realization of input-output nonlinear difference equation. In: *Proc. European Control Conference*, Brussels, 1997, Paper N 851.

[2] Sadegh, N . State realization of nonlinear systems described by input-output difference equations, In: *Proc. American Control Conference*, 1998, v. 6, 3344–3348.

[3] Kotta, Ü. and Tõnso, M. Transfer equivalence and realization of nonlinear higher order i/o difference equations using *Mathematica*. In: *Proc. Asia Pacific Conf. on Circuits and Systems: Microelectronics and Integration Systems*, Chiangmai, Thailand, 1998, 671–674.

SEMANTICS OF STATE-EVENTS IN HYBRID LANGUAGES

D.A. van Beek and J.E. Rooda
Eindhoven University of Technology, Department of Mechanical Engineering
P.O. Box 513, 5600 MB Eindhoven, The Netherlands
E-mail: d.a.v.beek@tue.nl, j.e.rooda@tue.nl

Abstract The semantics of a state-event statement is considered in a situation where one process changes the value of a variable, and another process executes a state-event statement – involving the same variable – at the same time-point. If the condition of the state-event statement is evaluated immediately, the shared variable may still have its old value. By means of an example model of a multi-section conveyor it is shown that immediate evaluation of state-event conditions is undesirable from a modelling point of view. The proposed semantics of the state-event statement is that the evaluation of the state-event condition is postponed until the other processes can no longer execute statements at the current time-point, and a new consistent state has been established. This semantics facilitates the development of correct models. The proposed semantics is analogous to the transition semantics in hybrid automata.

Introduction

Two main fields of research into hybrid systems are: formal analysis of hybrid systems, and simulation of hybrid systems. In the first field of research, model properties are derived by means of formal analysis. In the second field of research, simulation is used to obtain insight in dynamical system behaviour. The languages and tools used in the two fields are quite different. In the first field, formal languages are used: in many cases hybrid automata [1] or hybrid Petri nets. The languages and tools are designed to facilitate the construction of proofs. If simulation facilities are available, they are usually limited. Algebraic equations and algebraic loops, for example, are usually not allowed. In the second field on the other hand, simulation languages are used. Aspects that are considered important for simulation languages are among others: ease of use and expressive power of the language. A formal language semantics (meaning of the language elements) is, however, usually not available.

Concepts and techniques from both fields are required for the design of a hybrid simulation language. A first step on the way to achieve model correctness, is a formal definition of the language semantics. In the sequel, the semantics of language constructs for modelling of state-events is treated. Such a language construct is, for example, a *nabla* statement in χ [7], a *switch to* statement in gPROMS [2], and a *when* statement in Dymola [3]. In the sequel, we refer to the term state-event statement. By means of a model of a multi-section conveyor, the different possibilities for the semantics of state-event statements are explained. In this paper, a semantics is chosen that makes it easier for the modeller to develop correct models. The conveyor system is modelled using the hybrid χ language. The discrete-event part of the χ language is based on Communicating Sequential Processes (CSP), the continuous-time part on non-causal differential algebraic equations. In this paper, only a small subset of χ is used. Simulation models using the χ simulator have been used in a large number of industrial cases such as an integrated circuit manufacturing plant and a beer brewery. Verification of discrete-event χ processes is treated in [5]. Verification of hybrid χ models will follow in near future.

Model of a transport system

Figure 1 shows the transport system. The system consists of a line of conveyor belts driven by motors; it is used for transportation and buffering of boxes. Each conveyor belt is equipped with a sensor (represented in the figure by a small rectangle), that detects the presence of a box. Figure 2 shows the iconic structural χ model of two conveyor belts ($conv_0$ and $conv_1$) and the associated control processes (cc_0 and cc_1). The model is a highly simplified version of the model treated in [6]. The dashed lines with arrow heads represent the synchronization channels ($pc_0, pc_1, pc_2, p_0, p_1, p_2$), the lines ending in a small circle represent shared variables ($x_0, x_1, x_2, on_0, on_1, s_0, s_1$). The textual model is discussed below. They keyword type precedes the declaration of new types. The predefined type bool is the boolean data type.

type position = real, length = real, velocity = real, sensor = bool, actuator = bool

Systems are specified as follows: syst *name(parameter declarations)* = ⟦ *process and system declarations, channel and variable declarations* | *process and system instantiations* ⟧. System S is a textual specification of the iconic model from Figure 2. Two processes cc_0, cc_1 of type CC are declared, and two processes $conv_0$, $conv_1$

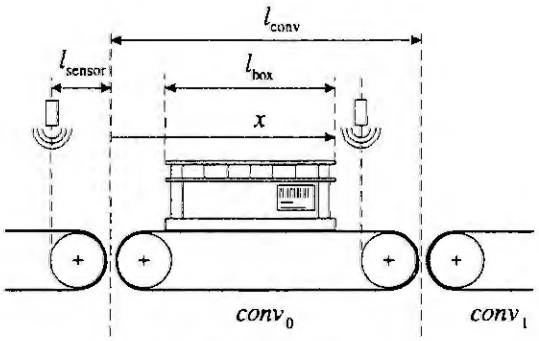


Figure 1: The transport system.

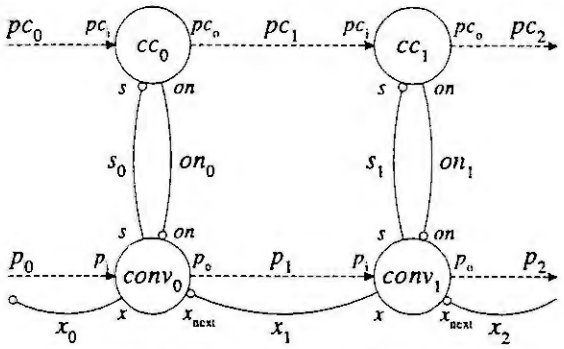


Figure 2: Model of the transport system.

of type *Conv* (Conveyor). Channels $pc_0, pc_1, \ldots, p_2$ are of type ~ void, which means that the channels are for synchronization only, and do not transmit data. Variables $x_0, x_1, x_2, on_0, on_1, s_0, s_1$ are declared in the system because they are shared between processes. Variables are either continuous or discrete. Continuous variables are declared using a double colon (e.g. x :: position); they are the unknowns in the equations. Discrete variables are declared using a single colon (e.g. on_0, on_1 : actuator); their value is determined by assignment statements only (e.g. *on* := false). Behind the bar |, the four processes cc_0, cc_1, $conv_0$ and $conv_1$ are instantiated with their actual parameters. The dots preceding and following the instantiations indicate that the additional conveyor belts preceding and following the two modelled conveyor belts are not shown in the specification.

```
syst S(l_conv, l_box, l_sensor : length, v_set : velocity) =
[[ cc_0, cc_1 : .CC, conv_0, conv_1 : .Conv, pc_0, pc_1, pc_2, p_0, p_1, p_2 : ~ void
, x_0, x_1, x_2 :: position, on_0, on_1 : actuator, s_0, s_1 : sensor
| ...
|| cc_0(pc_0, pc_1, on_0, s_0) || conv_0(p_0, p_1, x_0, x_1, on_0, s_0, l_conv, l_box, l_sensor, v_set)
|| cc_1(pc_1, pc_2, on_1, s_1) || conv_1(p_1, p_2, x_1, x_2, on_1, s_1, l_conv, l_box, l_sensor, v_set)
|| ...
]]
```

A process may consist of equations, discrete-event statements, or a combination of both: proc *name(parameter declarations)* = [[*variable declarations*; *initialization* | *equations* | *discrete-event statements*]]. Processes interact by means of channels and shared variables (continuous channels are no longer present in χ). Consider a channel p connecting two processes. Execution of p ~ in one process causes the process to be blocked until p ~ is executed in the other process.

The control process cc_0 (of type *CC*) shares variables on_0 and s_0 with process $conv_0$. The keyword xtern (external) preceding variables *on* and *s* in the parameter list of process *CC* indicates that these variables are defined externally (in this case, in system *S*). The process first switches the conveyor on (*on* := true). The remainder of the process is an infinite loop (*[...]). By means of a state-event statement ∇r, the discrete-event part of a process can synchronize with the equations of a process. Execution of ∇r causes the process to be blocked until relation r becomes true. By execution of statement ∇s, process cc_0 waits until s becomes true, which means that the box has reached the sensor position. After that, the conveyor $conv_0$ is switched off (*on* := false). Subsequently, process cc_0 tries to synchronize (pc_o ~) with process cc_1. When process cc_1 executes the corresponding synchronization statement (pc_i ~, since channel pc_o in cc_0 and channel pc_i in cc_1 are both connected to channel pc_1 in system *S*), conveyor $conv_0$ is switched on again (*on* := true). After the box has left the conveyor ($\nabla \neg s$, where $\neg$ means logical not, so that $\neg s$ is true when s is false), the loop is re-executed.

```
proc CC(pc_i, pc_o : ~ void, xtern on : actuator, xtern s : sensor) =
[[ on := true
| *[ pc_i ~; ∇s; on := false; pc_o ~; on := true; ∇¬s ]
]]
```

Processes $conv_0$ and $conv_1$ (of type *Conv*, see specification below) model the physical behaviour of the conveyor belts. Boolean variable *box* indicates the presence of a box on the conveyor. The equation of process *Conv* is a conditional equation: $[\, b_1 \longrightarrow equ_1 \,[]\, \ldots \,[]\, b_n \longrightarrow equ_n \,]$, where equ_i $(1 \leq i \leq n)$ represents an equation (or more than one equation). The boolean expression b_i denotes a guard. At any time, (at least) one guard must be open (true),

so that the equation associated with the open guard is activated. The position of the front of the box is modelled by means of an ordinary differential equation ($x' = v_{set}$ or $x' = 0$). When a box crosses the boundary of two conveyors $conv_0$ and $conv_1$, the box is transported from process $conv_0$ to $conv_1$ by means of a synchronization via channel p_1. From that point on, the position of the box will be registered in process $conv_1$, instead of process $conv_0$. Initially, there is no box on the conveyor and the sensor is off ($box := \text{false}$; $s := \text{false}$). When the box is present and the conveyor is switched on ($box \wedge on$), equation $x' = v_{set}$ is active. Here, x' denotes the time derivative of x, and v_{set} is the velocity of the conveyor belt. When the conveyor is switched off ($box \wedge \neg on$), equation $x' = 0$ is active. When there is no box on the conveyor ($\neg box$), there is no active equation. The meaning of $\neg box \longrightarrow$ undefined x, is that for as long as $\neg box$ is true (so that box is false), variable x is no longer an unknown in the equations, and is thus not defined by the equations; it behaves like a discrete variable. In the discrete-event part of $conv_0$, the process first synchronizes ($p_i\,^\sim$) with its predecessor to receive a box ($box := \text{true}$). The position of the box is initialized to 0. After that, the process waits until the box reaches the sensor position ($\nabla x \geq l_{conv} - l_{sensor}$), and switches the sensor on ($s := \text{true}$). When the box reaches the boundary between the two conveyors ($\nabla x \geq l_{conv}$), process $conv_0$ synchronizes with its successor process $conv_1$. When both processes are ready to synchronize (execution of $p_o\,^\sim$ in $conv_0$ and execution of $p_i\,^\sim$ in $conv_1$), the box is moved from $conv_0$ to $conv_1$ (execution of $box := \text{false}$ in $conv_0$ and execution of $box := \text{true}$; $x := 0$ in $conv_1$). Subsequently, process $conv_0$ waits until the rear of the box has passed the sensor, which happens when the position of the front of the box in the next conveyor (x_{next}) becomes equal to $l_{box} - l_{sensor}$ ($\nabla x_{next} \geq l_{box} - l_{sensor}$). After that, the sensor is switched off ($s := \text{false}$), and the loop is re-executed.

```
proc Conv( p_i, p_o : ~ void, xtern x, x_next :: position, xtern on : actuator, xtern s : sensor
         , l_conv, l_box, l_sensor : length, v_set : velocity
         ) =
|[ box : bool; box := false; s := false
| [ box ∧ on    ⟶ x' = v_set
  [] box ∧ ¬on  ⟶ x' = 0
  [] ¬box       ⟶ undefined x
  ]
| *[ p_i~; box := true; x := 0; ∇x ≥ l_conv − l_sensor; s := true; ∇x ≥ l_conv
   ; p_o~; box := false; ∇x_next ≥ l_box − l_sensor; s := false
   ]
]|
```

State-event semantics

To demonstrate the importance of the semantics of a state-event statement ∇r, the processes $conv_0$ and $conv_1$ are re-considered at the time-point when the box crosses the boundary between the conveyors. This is modelled by execution of the synchronization $p_o\,^\sim$ in process $conv_0$, and $p_i\,^\sim$ in $conv_1$. Just before the synchronization takes place, the value of variable box in process $conv_1$ is false (because there is no box present in that process), so that variable x not defined by an equation. The value of x can either be undefined (in the case that so far no box has entered process $conv_1$), or it can be equal to (or bigger than) l_{conv}, which is the value of x when the last box left the conveyor and box became false. Variable x_{next} in process $conv_0$ has the same value as variable x in process $conv_1$, since x_{next} in $conv_0$ and x in $conv_1$ refer to the same external variable x_1 in system S. Immediately after the synchronization has taken place, process $conv_0$ executes the statement $box := \text{false}$ (followed by $\nabla x_{next} \geq l_{box} - l_{sensor}$), and process $conv_1$ executes the statements $box := \text{true}$; $x := 0$. These statements are executed concurrently, which means that the execution order of the statements in process $conv_0$ relative to the execution order of the statements in process $conv_1$ is undetermined. Therefore, it is possible that first process $conv_1$ executes the statements $box := \text{true}$; $x := 0$, followed by the execution of statements $box := \text{false}$; $\nabla x_{next} \geq l_{box} - l_{sensor}$ in process $conv_0$. It is also possible that first process $conv_0$ executes the statements $box := \text{false}$; $\nabla x_{next} \geq l_{box} - l_{sensor}$, followed by execution of the statements $box := \text{true}$; $x := 0$ in process $conv_1$. In the last case, state-event statement $\nabla x_{next} \geq l_{box} - l_{sensor}$ is executed while variable x_{next} is either undefined or equal to (or bigger than) l_{conv}, instead of equal to 0. This example shows that it can be undesirable to evaluate the condition of a state-event statement immediately.

In order to ensure that state-event statements are executed when all processes have updated their variables, the proposed semantics of a state-event statement ∇r is that evaluation of the state-event condition r is postponed until the discrete-event parts of all processes are temporarily blocked in a synchronization statement, time-passing statement or state-event statement, and the unknown variables of the set of active equations have been solved. Such

a state is termed a 'consistent state'. This has the following implications for the conveyor example. When process $conv_0$ executes the statement $\nabla x_{next} \geq l_{box} - l_{sensor}$, evaluation of the condition $x_{next} \geq l_{box} - l_{sensor}$ is postponed until process $conv_1$ has become blocked in statement $\nabla x \geq l_{conv} - l_{sensor}$, and variable x_{next} is equal to 0.

Most models behave the same when the evaluation of a state-event condition is immediate, instead of postponed until the processes are blocked. There are, however, two reasons why immediate evaluation of a state-event condition is undesirable. First, it may lead to hard to find errors in models, because the correct execution of a model such as the conveyor system specified above, depends on the relative order of concurrently executing processes. If one process executes before the other, the model executes correctly; if the execution order is the other way round, the model executes incorrectly. Second, postponed execution of state-event conditions is analogous to the transition semantics of hybrid automata that are often used for formal analysis of hybrid systems [1]. In this way, the use of available tools and techniques for formal analysis of models using state-event statements is made easier. The analogy can be explained by the fact that the transition condition of a transition in a hybrid automaton is analogous to the state-event condition of languages with state-event statements. Furthermore, a transition t_{AB} from one state A of an automaton to another state B may be accompanied by assignments to variables. After a transition from state A to state B, new transitions from state B to other states are considered only when the assignments of transition t_{AB} have been executed.

In some modelling languages, connection elements are used instead of shared variables. E.g. streams in gPROMS [2] and terminals in Dymola [3]. In such languages, the variables x and x_{next} from the conveyor example would not be shared but would instead be connected to the same connection element. If the connection semantics is such that the connected variables become a single entity, a shared variable as treated above is realized. If the connection semantics is that of an algebraic equation denoting equality of the two variables, it needs to be specified when the equations are valid, and how equality is ensured after a discontinuous change to connected variables [4]. The results with regard to the proposed semantics of state-event statements, however, remain the same as in the shared variable case.

Conclusions

Postponing the evaluation of state-event conditions until the state of the processes is consistent facilitates the development of correct models, and corresponds to the semantics of hybrid automata. The relevance of this semantics has been illustrated by means of an example.

Acknowledgment

The authors like to thank Victor Bos for stimulating discussions about the language constructs and semantics of χ.

References

1. Alur, R., Courcoubetis, C., Halbwachs, N., Henzinger, T. A., Ho, P. H., Nicollin, X., Olivero, A., Sifakis, J., and Yovine, S., The algorithmic analysis of hybrid systems. Theoretical Computer Science 138, Springer, 1995, 3 – 34.

2. Barton, P. I. and Pantelides, C. C., Modeling of combined discrete/continuous processes. AIChE, 40 (1994), 966 – 979.

3. Elmqvist, H., Dymola – Dynamic modeling language – user's manual. Dynasim AB, Lund, 1994.

4. Fábián, G., van Beek, D. A., and Rooda, J. E., Semantics of model composition in hybrid languages. In: Proc. 1998 EUROSIM Conference, (Ed.: Juslin, K.) Helsinki, 1998, 269 – 276.

5. Kleijn, J. J. T., Reniers, M. A., and Rooda, J. E., A process algebra based verification of a production system. In: Proc. 2nd IEEE International Conference on Formal Engineering Methods (ICFEM'98), Brisbane, 1998, 90 – 99.

6. Van Beek, D. A., Gordijn, S. H. F., and Rooda, J. E., Integrating continuous-time and discrete-event concepts in modelling and simulation of manufacturing machines. Simulation Practice and Theory, 5 (1997), 653 – 669.

7. Van Beek, D. A. and Rooda, J. E., Languages and applications in hybrid modelling and simulation: Positioning of Chi. Control Engineering Practice, 8 (2000), 81 – 91.

FROM HUMAN-MACHINE-INTERACTION MODELING TO VIRTUAL AGENTS CONTROLING AUTONOMOUS SYSTEMS: A PHENOMENOLOGICAL ENGINEERING-ORIENTED APPROACH

Dirk Söffker
University of Wuppertal
Gaußstrasse 20, D-42097 Wuppertal

Abstract. The modeling of the Human-Machine-Interaction (HMI) gives ideas to transfer the understanding of human control and to apply the developed modeling technique to autonomous technical systems, like mobile robots. The task of this new kind of intelligent control is to respond autonomously and problem-equivalent to complex situations.

Introduction

Whereas classic control schemes typically address issues relating to speed, accuracies and other low-level problems, more complete theoretical models of system operation often are quite complex and unwieldy in unknown environments or situations.

In the sixties and seventies the human-control behavior was examinated for stimulus-response tasks, e.f. describing the time behavior of human driving etc., e.g. [1]. In the nineties the Human-Machine-Interaction is focussed. Different research directions have been established, which are oriented between Artificial Intelligence (AI) approaches and phenomenological macro cognition oriented engineering approaches [2]. In [3,4] a modeling approach of human interaction with a formalizable technical environment is developed. Core of the work is a specified situation-operator model (SOM). In contrast to known procedures [5,6,7] the developed approach is neither based on exact temporal logic nor assumes a mathematical perfect understanding of context structures. Gigerenzer and Goldstein [8] show that unsatisfying classical norms of rational inference fast and frugal algorithms can lead to effective rationality. The idea is to transfer the engineering oriented approach [3,4] to autonomous technical systems. In the sequence the SOM-technique is used to describe the 'human controler' (HC) as well as an autonomous system (AS) and will be summarized as intelligent system (IS).

From Scenes and Actions to Situations and Operators

Core of the approach is the assumption that the changes of the world are understood as a sequence of scenes and actions [3,4]. ISs are included in the real world (RW). Depending on the principal sensorial inputs, perceptions, and on the perception-defined knowledge base, the ISs adept and learn only parts of the RW. These parts are modeled using the special situation and operator calculus. The describable part of RW is called a system.

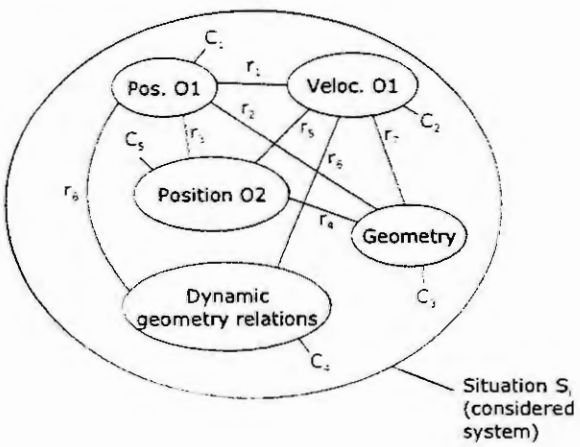


Fig. 1: Structure of the proposed item Situation S (Example)

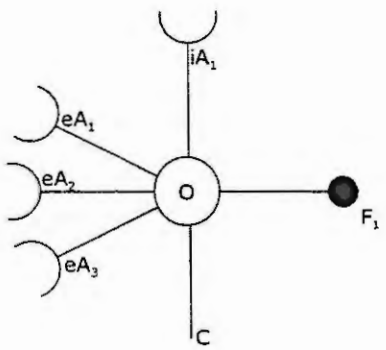


Fig. 2: Structure of the proposed item operator O

The item situation, which is (in contrast to [9]) a time-fixed system- and problemequivalent one, describes the system. The operator changes situations in time. The situation S consists of characteristics C and relations R. The characteristics are linguistic terms describing important facts (as qualities). This includes physical and informational values. The introduced item characteristic (C) includes the possibility of time-dependent parameters P. The relation R (of Cs) fixes the structure of the considered scene of the world modeled as situation S. The introduced situation concept allows the integration of different types of engineering-like descriptions.

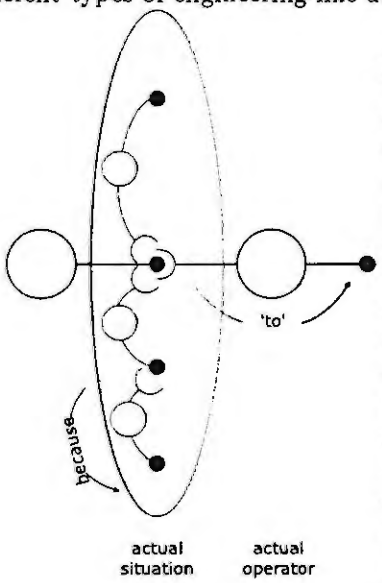


Fig. 3: Connection between situation and operator

A graphical illustration of the structure of a situation is given with fig. 1. The operator O (cf. fig. 2) is understood and modeled from a functional point of view: the operator is an information-theoretic construct which is defined by his function F (as the output) and assumptions. Here explicit and implicit assumptions eA, iA are distinguished. F will only be realized if the explicit assumptions eA are fulfilled. The iA include the constraints between the eA and F of the operator. The eA are of the same quality as the characteristics C of S. For the internal structure of the operator other descriptions like textual, logical, mathematical or other problem-related descriptions are allowed.
Operators are changing situations. This defines the discrete events of the change of situations. Operators and situations are strongly connected due to the identity of the characteristics of the situations and the explicit assumptions of the operators. This includes that the situation also consists of 'passiv' operators (internal causal relation: 'because'), whereby the change is done by 'active' operators (external causal relation: 'to'), illustrated with fig. 3. The change of the world results as a sequence of single actions modeled by operators, illustrated in fig. 4.

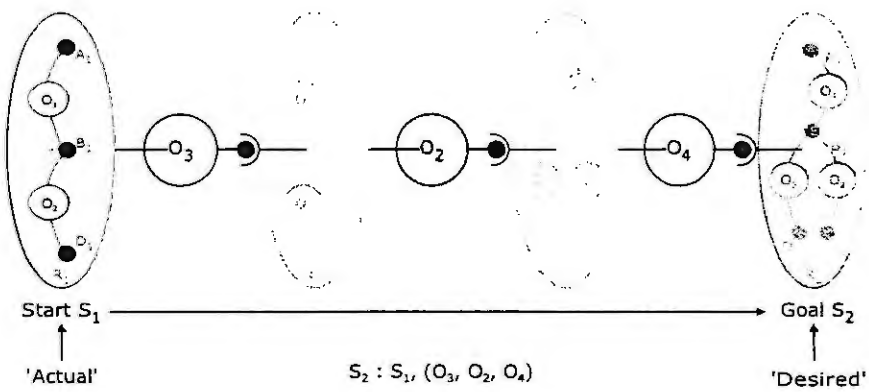


Fig. 4: Sequence of operators changing the situations from the originator to the desired goal

Please note that the operators correspond to situations. Both are not only used for structural organization of the (outerside) world of the system, but also for internal representation and storage. They are the core/background of all higher organized internal operations and functions of the IS like learning, planning etc. [3,4].

Learning

Following assumptions have been made:
• The problem-dependent structures of the real world scenes can be clearly identified as situation dependent R's and C's.
• The resulting identified S describe the RW in the way, that the relevant structure of the scenes and those of the identified S is equal.
• Operators are defined as time-independent.

Based on the introduced assumptions learning appears as the definition / redefinition of operators. This includes several different cases, where S_i denotes the i-th situation, R_i denotes the i-th relation and A_i, B_i, D_i denote the set of characteristics C_i of the i-th situation.

The classical straight forward learning includes the definition of O_1 by his induced (: active learning) or observed (: passive learning) situation changes,

$$O_i \quad : \quad S_i(R_i(C_i)) \rightarrow S_{i+1}(R_i(C_{i+1})) \quad R_i =' R'_{i+1}, C_i \neq C_{i+1} \tag{1}$$

$$O_i \quad : \quad S_i(R_i(C_i)) \rightarrow S_{i+1}(R_{i+1}(C_i)) \quad R_i \neq R_{i+1}, 'C_i = C'_{i+1} \tag{2}$$

$$O_i \quad : \quad S_i(R_i(C_i)) \rightarrow S_{i+1}(R_{i+1}(C_{i+1})) \quad R_i \neq R_{i+1}, C_i \neq C_{i+1} \tag{3}$$

which includes changes of situation structures $R_i \rightarrow R_{i+1}$, characteristics $C_i \rightarrow C_{i+1}$ or both.

This direct learning and definition procedure of operators is the main mechanism to map the outer world to the inner mental world of IS. This includes that IS is able to identify R_i, C_i from his sensorial inputs in combination with his actual knowledge. This cannot be assumed in general. To overcome the included problems of learning coincidentally coherencies and learning non-concretely coherencies due to unsufficient memory - mental model (MM) - capabilities, it is necessary to include backward oriented learning abilities: This includes the ability to distinguish Cs necessarily connected to R to those of coincidental presence not directly connected to the problem structure.

Example 1:

The reality consists of $S_i(R_i(A_i, B_i), D_i)$ (R_i connects A_i and B_i) and the learning mechanism of IS assumes/identifies S_i as $S_i(R_i(A_i, B_i, D_i))$ so O_1 will be learned as

$$O1 : S_1(R_1(A_1, B_1, D_1)) \rightarrow S_2(R_1(A_2, B_2, D_1)). \tag{4}$$

Due to the contingencies of the reality it may happen that

$$S_1(R_1(A_1, B_1)), O_1 \rightarrow S_2(R_1(A_2, B_2)) \tag{5}$$

can be observed, so IS gets the chance to rebuild the O_1 definition by 'replaying' to find the true $S_1, O_1 \rightarrow S_2$ sequence to redefine the objective O_1. In this way learning appears as a strongly nonlinear procedure due to the strong connection of the definition process of operators to the context, which includes the individual initial conditions of IS (the actual S and MM).

Example 2:

The task of IS should be the realization $S_1(R_1(A_1, B_1, D_1)) \rightarrow S_2(R_1(A_2, B_2, D_1)$. IS will take O_1, as learned (eq. 4). So the reality may be in contradiction to the MM, so there is a reason to reflect and change the definitions. It depends on internal features of IS to rebuild the MM immediately, after additional experiences or after extensive hypothesis oriented tests of definition of O_1.

Please note that this definitions of learning are independent from external commendations, penalties or rewards. The learning capabilities are the key feature for acting in unknown situations.

Planning, Action, and Achievement of planning

Planning is assumed as the mental preparation of the action or the series of actions to change actual S_{act} to desired ones $S_{des.}$, cf. fig. 4. Modeling of planning based on the SOM-technique includes a MM as a set of previously learned definitions / operators and the ability to identify the given goal $S_{des.}$ and $S_{act.}$. The goal elaboration is not considered here. Planning includes the elaboration of a ordered set of suitable O_i to solve the task $S_{act.} \rightarrow S_{des.}$. Due to the definition of S and O this can be done by comparison of C_i, eA, F applying a backward or forward reasoning strategy. The individual solvability strongly depends on the MM. If this cannot be solved exactly (different reasons possible), practical planning procedures are possible which use operators which do not exactly fulfill the requirements (full set of Cs), but requirements close to the desired perfect ones. This includes testing strategies, associative combinations (where internal similarities between the relation ea, F of the supposed unknown, but perfect O and the C of known O exist. In the reality conflicts appear between goals, part goals, necessary actions and reachable situations. Therefore decisions have to be made. This includes the development and evaluation of alternative paths (operator sequences), the choice of weighting factors etc. and also strongly depends

on the MM, and are not declared here.

The execution of the mentally prepared sequence of operators as actions realizes the interaction with RW. The interaction itself gives a variety of learning sequences: the result of each action of IS can be compared with the predicted one to optimize the internal MM etc.

The architecture of a SOM-based autonomous system

The architecture of the proposed behavior- and memory- based open system is given in fig. 5. Here additionally modules of hypothesis testing, of data based reconfiguration and, very important, of object- and scene recognition, scene and phenomenological interpretation are given for the example of a mobile robot. It will be clear that the system is able to get a perfect impression of his environment, depending on his sensoral inputs, the observable change of the environment and on his interaction. The system is designed for unknown environment, which includes that firstly interactions are needed to build the MM.

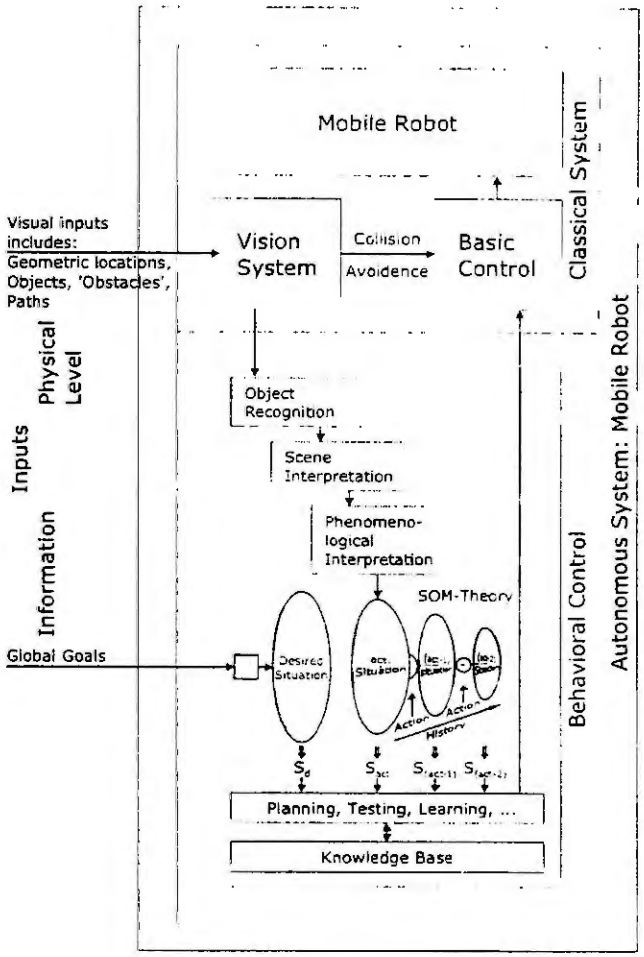


Fig. 5: Outline of the proposed intelligent control scheme based on the SOM-technique

Summary

The contribution briefly describes elements and architecture of a new kind of intelligent control to autonomous systems like mobile robots based on the description of the Human-Machine-Interaction.

References

1. Schweitzer, G.: Probleme und Methoden zur Untersuchung des Regelverhaltens des Menschen. In: Oppelt, W.; Vossius, G.: Der Mensch als Regler. VEB Verlag Technik, Berlin, 1970, 159-238.
2. Cacciabue, P.C.: Modelling and Simulation of Human Behavior in System Control. Springer series: Advances in Industrial Control, Springer, Berlin, Heidelberg, New York, 1998.
3. Söffker, D.: Human Control. Habilitation Thesis (in german), in preparation.
4. Söffker, D.: Modeling the Human-Machine Interaction. in: Technical Report SS-99-04 Knowledge System Lab, Stanford University, California, 1999.
5. Görz, G. (Hrsg.): Einführung in die künstliche Intelligenz. Addison-Wesley, 1995.
6. Müller, J. (Hrsg.): Verteilte Künstliche Intelligenz. BI-Wissenschaftsverlag, 1993.
7. Sandewall, E.: Features and Fluents. Clarendon Press, Oxford, 1994.
8. Gigerenzer, G.; Goldstein, D.G.: Reasoning the Fast und Frugal Way: Models of Bounded Rationality. Psychological Review 103, 1996, No. 4, 650-669.
9. McCarthy, J.; Hayes, P.J.: Some Philosophical Problems from the Standpoint of Artificial Intelligence. Machine Intelligence, 4, 1969.

MODELING PROBABILITY DISTRIBUTIONS FROM DATA AND ITS INFLUENCE ON SIMULATION

Wolfgang Hörmann and Onur Bayar
Bogazici University Istanbul
80815 Bebek-Istanbul, Turkey

This work was supported by the Austrian Academy of Science, APART scholarship.

Abstract. Generating random variates as generalisation of a given sample is an important task for stochastic simulations. The three main methods suggested in the literature are: fitting a standard distribution, constructing an empirical distribution that approximates the cumulative distribution function and generating variates from the kernel density estimate of the data. The last method is practically unknown in the simulation literature although it is as simple as the other two methods. The comparison of the theoretical performance of the methods and the results of three small simulation studies show that a variance corrected version of kernel density estimation performs best and should be used for generating variates directly from a sample.

Introduction.

It is well known that the choice of the input distribution is a crucial task for building a stochastic simulation model. If the inputs of the real system we are interested in are observable, it is possible to collect data. In this case the choice of the input distribution for the stochastic simulation model is a statistical problem, which can be called the modelling of probability distributions from data. The problem can be solved in a parametric approach by estimating the parameters of a suitable standard distribution or in a non-parametric approach by estimating the unknown distribution. We are convinced that due to its greater flexibility the non-parametric approach should be used unless there are profound a priori reasons (eg. of physical nature) favouring a certain standard distribution.

In stochastic simulation we are interested not only in estimating the input distribution but also in generating random variates from that distribution. This task is called "generating variates from empirical distributions" or "generalising a sample" in the simulation literature (see eg. [1] and [5]). As these names indicate, the problem of estimating (or modelling) the input distribution is often hidden behind a procedure to generate random variates from data. Perhaps that is the reason that no comparison of the quality of the estimation of the different methods was done till now, allthough there is a developed statistical theory discussing the optimal estimation of densities. Especially kernel density estimation is well suited for modelling input distributions, as variate generation from these estimates is very simple. This was already observed in the monographs [4], [2] and [6] but seems to be widely unknown in the simulation literature.

Therefore this paper compares the theoretical properties of these different methods of generating random variates from data and will demonstrate with simple examples that the choice of the method can have an influence on simulation results.

Sampling from Empirical Distributions

We are given a random sample of size n, denoted by $X_1, X_2, \ldots, X_n$. s will denote the sample standard deviation. Of course the simplest method of sampling from the empirical distribution is **naive resampling**. We just take randomly numbers of the sample. If the sample is based on a continuous random variable this method has the obvious drawback, that only a small number of different values can be generated.

To overcome these problems two well known simulation text-books ([1] and [5]) suggest to use a linear interpolation of the empirical cumulative distribution function (CDF) for generating random variates. The algorithm suggested in [5] (we shall call it **ELK** in the sequel) is only generating points between the minimum and maximum of the sample, whereas the algorithm suggested in [1] (called **EBFS**) in this paper) uses an exponential tail on the right hand side of the sample. Both algorithms are simple to implement.

There is another simple adaptation of naive resampling called smoothed bootstrap in the statistic literature. Do not only resample but add to any of the resampled numbers some noise, ie. a continuous

random variable with 0 expectation and small variance. It is not difficult to see, that smoothed bootstrap is the same as generating random variates from a density-estimate by using the kernel method, but it is not even necessary to compute the estimated density.

Algorithm **KDE**: (Kernel Density estimation)
(0) Set-up: Choose the smoothing parameter b (see below for the formula).
(1) Generate a random integer I uniformly distributed on $(1, 2, \ldots, n)$
(2) Generate a random variate W from the noise distribution
(3) Return $Y = X_I + bW$

The density of the random noise distribution W is called kernel and will be denoted by $k(x)$. Clearly $k(x)$ must be a density function and should be symmetric around the origin. As we want to change the variance of the random noise we introduce the scale parameter b (called bandwidth or smoothing parameter in density estimation); the random variable bW has the density $k(x/b)/b$. The random variate Y generated by Algorithm KDE is the equiprobable mixture of n noise distributions, each centered around one of the sample points. This implies that the density of Y (denoted f_Y) is the sum of n translated versions of $k(x)$ multiplied with $1/n$. f_Y is the kernel density estimate of the unknown distribution and is called $\hat{f}$ in the literature.

$$f_Y(x) = \frac{1}{nb} \sum_{i=1}^{n} k\left(\frac{x - X_i}{b}\right)$$

Of course there remains the question of the choice of the bandwidth b and the kernel function $k(x)$. Here we can use the results of the theory of density estimation as presented eg. in [6] or [7]. To minimise the mean integrated squared error we use a very simple and robust variant of estimating the optimal bandwidth b as given in [6]. $b = \alpha(k)\, 1.364\, \min(s, R/1.34)\, n^{-1/5}$,
where the constant $\alpha(k)$ is 0.776 for the Gaussian and 1.351 for the rectangular kernel respectively. s denotes the standard deviation and R the interquartile range of the sample. There are lots of much more complicated ways to determine b published in literature. For an overview see [3], where the L_1-error (ie. the mean integrated absolute error) of many different bandwidth selection procedures is compared. The method we use is a mixture of the methods called "reference: L_2, quartile" and "reference: L_2, std. dev" in [3]. The results of the simulation study show that with the exception of some very strangely shaped multimodal distributions the performance of this very simple choice of b is not bad. And we are not interested in an optimal estimation of the density here but in constructing an empirical distribution that is "as close as possible" to the theoretic distribution in all aspects.

The last question that has to be solved before we can use Algorithm KDE is the choice of the kernel. Asymptotic theory shows that the MISE is minimal for the Epanechnikov kernel $f(x) = (1 - x^2)3/4$ but some other kernels have allmost the same efficiency. Therefore we can choose the kernel by also considering other properties, eg. the speed and simplicity of our generation algorithm. In that respect the rectangular kernel (ie. uniformly distributed noise) is of course the best choice, but it has the theoretical draw-back that the estimated density is not continuous. Due to the nice statistical interpretation we prefer Gaussian noise and will use it in the sequel.

Algorithm KDE guarantees that the density function of the empirical distribution approximates the density of the unknown true distribution as good as possible with respect to the mean integrated squared error. On the other hand we clearly see, that for algorithm KDE the variance of the empirical distribution is always larger than the variance of the observed sample. This can be a disadvantage in simulations that are sensitive against changes of the variance of the input distributions. To overcome this problem it is possible to force the empirical distribution to have the same variance as the sample in the following way (suggested in [6]).

Algorithm **KDEVC**:
(0) Set-up: Compute the mean $\bar{x}$, the standard deviation s and the interquartile range R of the sample. Compute $b = \alpha(k)\, 1.364\, \min(s, R/1.34)\, n^{-1/5}$
(1) Generate a random integer I uniformly distributed on $(1, 2, \ldots, n)$
(2) Generate a random variate W from the noise distribution
(3) Return $Y = \bar{x} + (X_I - \bar{x} + bW)/(1 + b^2\sigma_k^2/s^2)^{1/2}$ (σ_k^2 denotes the variance of the kernel)

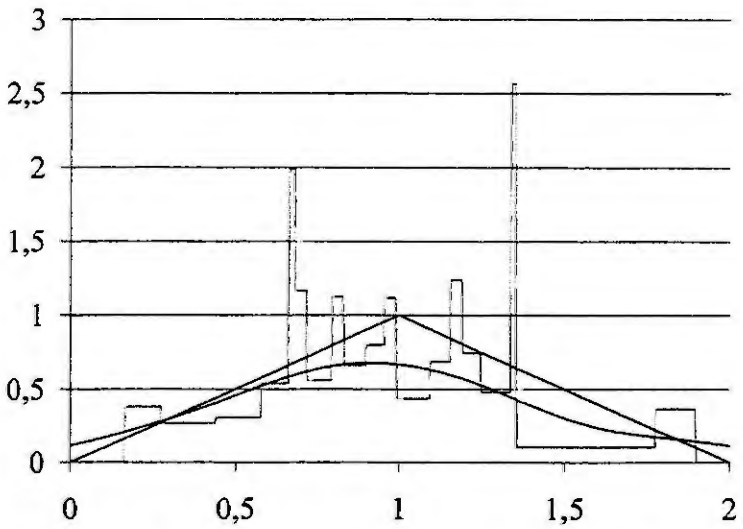


Figure 1: A triangular density with the empirical densities of ELK (step function) and KDE

Remark: Positive random variables are interesting for many applications. Method KDE can cause problems for such applications as it will also generate negative variates. The easiest way out is the so-called mirroring principle. Instead of a negative number Y simply return $-Y$. Unfortunately the mirroring principle disturbs the variance correction. They can be used together but the resulting empirical distribution has a smaller variance than the sample. This can only be a practical problem if the sample of a positive distribution has many values close to zero.

Comparison of Methods

Expectation and Variance

An important concern for many simulations is the expectation and the variance of the input distribution. The best we can hope to reach is that the expectation and the variance of the empirical distribution are equal to the sample mean and sample variance of the observed sample as these are the best available estimates for the unknown values. All methods described above produce random variates that have as expectation the sample mean. Only for ELK the result is slightly different.

Concerning the variance the situation is more complicated: For kernel density estimation we know that the variance of the empirical distribution is larger than the sample variance. A simple calculation shows that $V(KDE) = s^2((n-1)/n + b^2\sigma_k^2)$ which is $s^2(1 - 1/n + 1.058\,n^{-2/5})$ for the Gaussian kernel and our choice of b. For eg. $n = 100$ the variance factor $V(\text{emp. distr})/s^2 = 1.164$) which shows that it may be wise to consider the variance corrected version of the algorithm KDEVC which has by design a factor of one for any sample size. A second possibility to reduce the variance factor of Algorithm KDE is the use of a smaller bandwidth b. For the limiting case $b \to 0$ Algorithm KDE coincides with naive resampling and thus has a variance factor of $(n-1)/n$.

For the two methods based on linear approximation of the CDF (ELK and EBFS) there is no simple formula for the variance of the empirical distribution as the variance depends on the sample. So we computed the variance of the empirical distribution in a small simulation study. Our results show that for ELK the variance of the empirical distribution is always smaller than the sample variance, for EBFS the variance is due to the added tail always bigger. The factor $V(\text{emp. distr})/s^2$ is strongly influenced by the shape of the theoretic distribution. For samples of size 100 we observed factors up to 1.12 for EBFS and down to 0.91 for ELK.

The fine structure of the empirical distribution

Thanks to L. Devroye there exists a theoretical result about the quality of the local approximation of the unknown density by the methods ELK and EBFS. He showed (see [1] p. 132) that for n towards

infinity the density of the empirical distribution does not even converge against the correct distribution. In contrast to this poor behaviour we know that for method KDE the estimated density converges and we even have (approximately) minimised the mean integrated squared error. For method KDEVC the optimal approximation of the density is slightly disturbed by the correction of the variance. Nevertheless we know that asymptotically KDEVC has very nice approximation properties because it coincides with KDE. As this theoretical argument seems to be unimpressive for simulation practitioners we try to illustrate the consequence of this theoretical result in Figure 1. It compares for a "well behaved" sample of size 20 (from a triangular distribution) the empirical density of method ELK (which is practically identical with EBFS) and of method KDE. Looking at Figure 1 we can also understand that the high peaks in the ELK-density occur when two sample points are comparatively close together and this happens in practically all samples.

The distance between the theoretical and the empirical distributions

There are different distance measures suggested in the literature to compute the distance between two distributions. In density estimation the L_2-difference (integrated squared difference) and the L_1- difference (integrated absolute difference) are of major importance. Another possibility is to use the L_1-difference or the L_2-difference between the two CDFs. As method KDE is based on estimating the density whereas ELK and EBFS are based on an approximation of the CDF we thought that these four measures would favour automatically one group of the algorithms discussed here. Therefore we decided to use a third class of distance measures for our comparison: The test-statistics of three well known goodness-of-fit tests, the Chi-square test, the Kolmogorov-Smirnov test and the Anderson-Darling test. For the Chi-square test we took the number of equiprobable classes as $\lfloor\sqrt{m}\rfloor$. Then the test-statistic divided through m converges against $\int(\hat{f}(x)-f(x))^2/f(x)dx$ a weighted squared difference of empirical and theoretical density. ($\hat{f}$ denotes the density of the empirical distribution of the different methods.) These considerations clearly show that the chi-square test is more sensitive to deviations in the tails than to deviations in the centre of the distribution. The Kolmogorov-Smirnov test measures the maximal absolute difference between the empirical CDF and the theoretical CDF. Its test statistic is $D_m = \sup(|F_m(x) - F(x)|)$ (where F_m denotes the emprical CDF of the sample). The power of the KS-test for deviations in the tails is low. The Anderson-Darling test on the other hand was designed to detect discrepancies between the tails of the distributions. Like the KS-test it compares the theoretical and the empirical CDF but it uses a weighted L_2-difference to compare the CDFs. The test statistic is $A^2_m = m\int(F_m(x) - F(x))^2 f(x)/(F(x)(1-F(x)))dx$

Then for each random sample of size $n = 100$ of the different theoretical distributions and for each of the five different methods we generated 40 different samples of the empirical distributions with size $m = 3000$. The computed the average test statistics for all these experiments and repeated them for 40 different samples of the theoretical distribution.

Table 1 gives the final average of the different test staistics. These results can be seen as a (stochastic) distance measure between the theoretical distribution and the empirical distribution. They are (like eg. the mean intagrated squared error) a measure for the deviation between empirical and theoretical distribution averaged over different samples from the theoretical distribution. We can see that all averages are in the critical region of the respective tests. This is not surprising. As the empirical distribution is based on a sample of size 100 only, a much larger sample ($m = 3000$) from the empirical distribution cannot have exactly the same properties as a sample from the correct distribution. In the chi-square and in the Kolmogorov-Smirnov tests methods KDE and KDEVC perform considerably better than EBFS, ELK and naive resampling. For the Anderson-Darling tests the differences are small but even there KDEVC performs best. The results do not only show that kernel density estimation performs better than the other methods, they also show that the variance corrected method performs in almost all cases better than the original version. We were astonished that for these three very different distance measures and for four quite different distributions the same method for constructing the empirical distribution is best or close to best in all cases. We think that this result is a strong argument in favour of method KDEVC.

We added the last column of Table 1 to compare the discussed methods with the method of fitting a standard distribution (FSD). Of course FSD performs best if we fit the correct distribution but Table 1 shows that KDEVC is in most cases not far away which means that we do not loose much in using KDEVC instead of a standard distribution. The two mixture distributions were chosen such that their shape is

not far away from a standard distribution. If we assume – as we do for FSD – that the data come from a standard normal distribution with unknown parameters, we would estimate the parameters μ and σ and then conduct a chi-square test. The power of the test (the probability to reject the hypothesised normal distribution) is 0.5 if the unknown true distribution of the sample is our normal equiprobable mixture of N(0,1) & N(3,1) and the sample size is 100. Thus the poor results for fitting a normal distribution to the normal mixture are not artificial numbers. They have practical relevance as the power of the goodness-of-fit tests is often too low to show the deviation from the hypothesised standard distribution. For the gamma mixture we used a distribution, which is even closer to a gamma distribution. Only for 15 % of all samples of size 100 the chi-square test rejects the hypothesised gamma distribution. The distance measures show that the quality of the approximation of FSD and KDEVC is about the same. Of course it is no problem to find examples where FSD performs arbitrarily poor. Just take a theroetic distribution with a shape far away from any standard distribution.

Table 1: Average Test Statistics and standard errors (in brackets).

	Critical value 5%	EBFS	ELK	Naive resampling	KDE	KDEVC	Standard distribution
Theoretical distribution: Gamma(2)							
χ^2-mean(SE)	71.0	1218 (24)	1270 (25)	1642 (31)	319 (10)	241 (7)	110 (6)
KS-mean*1000	24.8	79 (2)	79 (2)	84 (2)	56 (2)	54 (2)	44 (2)
AD-mean	2.5	27 (2)	27 (2)	27 (2)	31 (2)	24 (2)	17 (2)
Theoretical distribution: Gamma mixture: G(2)&G(6)							
χ^2-mean(SE)	71.0	1196 (23)	1256 (24)	1651 (28)	268 (6)	230 (6)	220 (5)
KS-mean*1000	24.8	84 (3)	84 (3)	88 (3)	57 (2)	63 (2)	68 (2)
AD-mean	2.5	32 (2)	32 (2)	32 (2)	30 (2)	28 (2)	628 (2)
Theoretical distribution: Normal(0,1)							
χ^2-mean(SE)	71.0	1290 (26)	1240 (23)	1633 (30)	220 (12)	155 (7)	113 (7)
KS-mean*1000	24.8	82 (2)	83 (2)	87 (2)	59 (2)	54 (2)	46 (2)
AD-mean	2.5	30 (2)	30 (2)	31 (2)	30 (2)	22 (2)	19 (2)
Theoretical distribution: Normal Mixture: N(0,1)&N(3,1)							
χ^2-mean(SE)	71.0	1261 (29)	1229 (26)	1601 (33)	350 (18)	209 (9)	513 (7)
KS-mean*1000	24.8	80 (3)	80 (3)	80 (3)	62 (2)	65 (3)	92 (2)
AD-mean	2.5	32 (3)	31 (3)	31 (2)	35 (3)	26 (2)	44 (2)

Influence on Simulation results

Changing the method of modelling the empirical distribution is not more than changing the fine structure and perhaps slightly the variance of the input distribution of a simulation model. It is to be expected that many simulations, which have as ouput averages of a large number of input random variables, are not very sensitive to small changes in the fine structure of the input distribution. For example it is known that the average waiting time in the M/G/1 queue is only influenced by the expectation and the variance of the service time distribution and not by its shape. And it is even better known that the distribution of the sample mean of a large sample is always very close to normal. The parameters of that normal distribution are again only influenced by the expectation and the variance of the underlying distribution and not by its shape. These are arguments why the choice of the method will not have a big influence on many simulation results. Nevertheless we try to get some insight into this question by looking at three examples. The first simulation model we tried is the M/G/1 queue. The inter-arrival times are taken exponential with expectation 1, the service times are modelled from samples of different gamma distributions, using the different empirical methods described above. Then we simulated the model starting with an empty system and observed the average waiting time (AVW) and the maximal number in queue (MAXNIQ). We repeated this experiment for several different samples of the theoretical service-time distribution to get an average over different samples. We assumed in advance that this model is probably very stable with respect to small changes of the fine structure of the distribution but we tried it because of its importance and because we thought that the tail-modelling of the empirical distribution could have some influence on the results. The results given in Table 2 mainly show that there is little to choose between the different methods to fit an empirical distribution, all methods have about the same performance and rarely differ more than one standard error. The second interesting result is that the

size of the error when using an empirical instead of the correct distribution strongly depends on ρ, the utilisation factor of the system. The results for $\rho = 0.4$ and $n = 100$ are better than those for $\rho = 0.9$ and $n = 500$.

Table 2: M/G/1-queue: Average Error and its Standard Error (in brackets)

	ρ	a of Γ distr.	n	KDE	KDEVC	EBFS	Naive Resampling	ELK
AVW*100	0.9	10	100	137(21)	137(21)	138(21)	138(21)	132(19)
MAXNIQ*100	0.9	10	100	340(41)	341(40)	341(42)	343(40)	332(37)
AVW*100	0.9	10	500	53(4)	52 (4)	52(4)	52(4)	53(4)
MAXNIQ*100	0.9	10	500	142(11)	137(10)	138(11)	140(11)	143(10)
AVW*100	0.4	2	100	3(0.3)	3(0.3)	3(0.3)	3(0.2)	3(0.2)
MAXNIQ*100	0.4	2	100	37(3)	37(3)	47(4)	36(3)	38(3)
AVW*100	0.4	2	500	1.5(0.1)	1.5(0.1)	1.5(0.1)	1.5(0.1)	1.5(0.1)
MAXNIQ*100	0.4	2	500	18(1)	19(1)	21(2)	17(1)	18(1)
AVW*100	0.4	10	100	1.0(0.1)	1.0(0.1)	1.0(0.1)	1.0(0.1)	1.0(0.1)
MAXNIQ*100	0.4	10	100	12(1)	11(1)	13(1)	11(1)	12(1)
AVW*100	0.4	10	500	0.5(0.03)	0.5(0.03)	0.5(0.03)	0.5(0.03)	0.5(0.03)
MAXNIQ*100	0.4	10	500	5(0.4)	6(0.4)	6(0.4)	5(0.4)	5(0.4)

Due to the very small differences between the methods for the M/G/1-queue we looked for simulation examples that are influenced by the fine structure of the distribution. So we tried the following: We take a sample of size 50 of a gamma distribution and compute the maximal and the minimal distance between two neighbouring points. What happens in that experiment if the gamma distribution is replaced by an empirical distribution constructed from a sample of size $n = 100$ or 500 of the correct gamma distribution? We repeated each experiment 10000 times and arrived at the results given in Table 3.

Table 3: Average minimal and maximal distances, (standard errors in brackets)

	n	EBFS	ELK	Naive resampling	KDE	KDEVC	Correct distrib.
Theoretical distribution: Gamma(2)							
min*105 (SE)	100	55(0.7)	54(0.7)	0 (0)	172(2)	168(2)	163(2)
min*105 (SE)	500	76(1)	77 (1)	12 (0.6)	173(2)	167(2)	163(2)
max *100 (SE)	100	196 (2)	125 (1)	147 (1)	138(1)	129(1)	150(1)
max *100 (SE)	500	172 (2)	139 (1)	148 (1)	146(1)	141(1)	150(1)
Theoretical distribution: Gamma(20)							
min*105 (SE)	100	209 (3)	199(3)	0 (0)	679(7)	624(6)	628(6)
min*105 (SE)	500	306 (4)	304 (4)	53 (3)	662(7)	628(6)	628(6)
max *100 (SE)	100	706 (5)	291 (1)	342 (2)	339(2)	310(2)	323(2)
max *100 (SE)	500	482 (4)	314 (2)	331 (2)	338(2)	324(2)	323(2)

The interpretation of Table 3 with respect to the minimal distance is simple. Naive resampling is useless if the fine structure of the distribution is of any importance, even though the sample generated from the empirical distribution had only size $m = 50$ whereas $n = 100$ or even $n = 500$ data points were available. The second observation is that the fine structure of EBFS and ELK are only slightly better whereas those of KDE and KDEVC are much better with results close to the results using the correct distribution. Interesting is the fact that the results of the variance corrected method are better than those of the standard method. If we look at the results for the maximal distance we see that naive resampling works better than expected. The results of EBFS are worse than expected, although EBFS assumes exponential tails, which should be an advantage. KDE and KDEVC again show good results.

Our last example can be interpreted as part of a computer-system simulation. Two processes work with the same file. They start at the same time and the time between two file-accesses follows the same distribution (gamma(10, 0.1)). Now we want to estimate the probability that the two processes try to access the file at "almost the same time", ie. that the time difference is smaller than a given tolerance. What happens in this example with the simulation results if again the gamma distribution is replaced by an empirical distribution which is constructed from a sample from the correct distribution? Our results are given in Table 4. As we have observed in Table 3 the empirical distributions constructed by KDE and KDEVC have about the same behaviour as the correct distribution. EBFS and ELK are considerably worse whereas naive resampling is totally useless for this example.

Table 4: Estimated probability of file access at the same time, (SE of estimate in brackets)

	n	tol	EBFS	ELK	Naive resampling	KDE	KDEVC	Correct distrib.
Prob*10^6(SE)	100	10^{-5}	306 (25)	282 (24)	10328 (143)	202(20)	240 (22)	167 (10)
Prob*10^6(SE)	500	10^{-5}	242 (22)	248 (22)	2156 (66)	186(19)	214 (21)	167 (10)
Prob*10^6(SE)	100	10^{-4}	2592 (72)	2628 (72)	12142 (155)	1960(63)	1978(63)	1898 (30)
Prob*10^6(SE)	500	10^{-4}	2278 (67)	2234 (67)	3956 (89)	2044(64)	2026(64)	1898 (30)

Future Work

It is also possible to use kernel functions that have heavier tails than the normal distribution, for example the density of the t-distribution or of the logistic distribution. Allthough not used in density estimation they could be interesting for our purpose as they allow to generate distributions with the same behaviour as the given sample but different tail behaviour. They could be used in simulation studies to test the influence of the tails of the input distribution on the final results. We tried as kernels the Gaussian, the uniform, the logistic and the t-distribution (with 3 degrees of freedom). There were clear differences between different used kernels in the results of the maximal distance in Table 3, which is obviously sensitive to the tail behaviour of the input distribution. As the results for all other tables were practically the same for all different kernels we have only reported the results of the Gaussian kernel. Nevertheless we think that the use of different (heavier tailed) kernels in simulation studies would deserve future discussion. An additional advantage of the kernel method is the possibility to generalize it to higher dimensions. This is important as with the exception of the normal distribution few standard distributions are commonly used to model multivariate data. We will present the details in a subsequent paper.

Conclusions

The first of the final conclusions from the above investigations is in our opinion that methods that generate random variates directly from data are important and useful tools in simulation studies. They are easy to use and more flexible than fitting standard distributions to data. They should be used whenever there are no a priori reasons for using a certain standard distribution. The second conclusion is even more obvious. Use kernel density estimates to construct the empirical distribution function. Allthough the question which variant should be taken is not fully solved here, we think that the results presented in this paper clearly favour the variance corrected version (KDEVC) allthough one could find applications where the original version KDE performs better.

Sampling from kernel density estimates is a simple task. There is mathematical theory that shows the good theoretical behaviour of these estimates, and the empirical results of this paper confirm that these good theoretical properties can lead to more accurate results in simulation studies. Thus it is an important tool for modelling input distributions in simulation studies.

References

[1] P. Bratley, B. L. Fox, and E. L. Schrage. A Guide to Simulation. Springer-Verlag, New York, 2 edition, 1987.

[2] L. Devroye. Non-Uniform Random Variate Generation. Springer-Verlag, New-York, 1986.

[3] L. Devroye. Universal smoothing factor selection in density estimation, theory and practice. Test, 6 (1997), 223–320.

[4] L. Devroye and L. Györfi. Nonparametric Density Estimation: The L_1 View. John Wiley, New-York, 1985.

[5] A. Law and D. Kelton. Simulation Modeling and Analysis. Mc-Graw-Hill, New-York, 1991.

[6] B. Silverman. Density Estimation for Statistics and Data Analysis. Chapman and Hall, London, 1986.

[7] M. Wand and M. Jones. Kernel Smoothing. Chapman and Hall, London, 1995.

HIERARCHICAL DISCRETE-EVENT MODELS OF CONTINUOUS SYSTEMS

Patrick Philips and Heinz A. Preisig
Systems and Control Group, Eindhoven University of Technology
P.O.Box 513, Eindhoven, The Netherlands
e-mail: p.p.h.h.philips@tue.nl

Abstract. Discrete-event representations (models) of continuous systems are frequently arising in systems and control theory. A disadvantage of the discretization of continuous plants is the computational effort which is necessary to obtain these models. In this paper we will follow an approach to reduce the number and problem-size of the optimizations that are involved in the discretizing method. For this, the system is divided into sub-models. It is shown that by computing in parallel the state-transitions of sets of coordinates, each set associated with a sub-model, the transitions of the overall model are computed. At the same time, state agglomeration is more likely to be successful for each of the sub-models than for the overall system. The computational benefit is illustrated by means of an example.

Introduction

Discrete-event representations (models) of continuous systems are frequently arising in systems and control theory. For example, a system that is continuous by nature but is only observed by discrete sensors, can be represented by a discrete model. Also the analysis of some hybrid systems can be facilitated by transforming the continuous part into a discrete-event part such that only a discrete-event system has to be investigated which may be less difficult than studying the underlying hybrid system. A typical situation is when a (possibly controlled) continuous plant is to be supervised by some programmable logical controller or computer program which uses discrete state information. Since the discrete controller cannot communicate with the system at a continuous level it is necessary to use an interface. This corresponds to discretizing the continuous state space of the system into a finite set of symbols to be used by the controller. Various formalisms and methods have been proposed to describe these kind of systems and to solve the discretizing problem.

In [1] a qualitative modelling approach for linear, discrete-time systems is presented and necessary and sufficient conditions are given for which the resulting automaton is deterministic. The discretizing method proposed in [5] is able to deal with non-linear systems and is based on a test which has similarities with the method for linear systems presented in [3], which is extended for nonlinear systems in [2]. We focus on the latter procedure.

A disadvantage of the state discretization of continuous plants is the computational effort which is necessary to obtain these models. The underlying combinatorial growth characteristic is known as the state-explosion problem. In this paper we will follow an approach for reducing the number of optimizations and the dimensionality of each individual optimization problem involved in the discretizing method. The core idea of the method is to build the model of the complete plant from models of parts thereof. Of course, these sub-models have to provide the same information as is stored in the complete model. Another advantage of using these sub-models is that they provide a way to reduce the 'size' of such sub-model, such that the computations involved in the usage of the discrete model (e.g. prediction of the next discrete state) also can be reduced.

The paper is organized as follows. First the discrete-event modelling algorithm is explained. Next, the computational effort necessary to perform the algorithm is discussed. Then, it is explained how to reduce the number of computations by exploiting the sparsity of the system and creating a hierarchical structure. How to reduce the number of states for a model is discussed next. Finally, an example is given and some conclusions are drawn.

Discrete-event modelling algorithm

Suppose we are given a continuous system described by a set of differential equations in $\mathbb{R}^n$:

$$\dot{x} = f(x,u),\ x \in \mathbb{R}^n, u \in \mathbb{R}^m \tag{1}$$

It is assumed that f is continuous and that the system (1) has a unique solution for every initial value in the region of interest.

The discrete-event model, which has to result, is given by the system [4]:

$$\Sigma = (\tilde{X}, \tilde{U}, \phi)$$

First, we define a set of discrete states $\tilde{X}$ and discrete inputs $\tilde{U}$ for the discrete event dynamic system to be constructed. Then, the transition function ϕ is determined.

Discrete states: For each coordinate x^i, $(i = 1, ..., n)$, we choose a set of boundaries:

$$\beta_0^i < \beta_1^i < \ldots < \beta_{\tilde{n}_i}^i \quad (\tilde{n}_i \geq 1). \tag{2}$$

We can think of the state-space being partitioned into hypercubes (cells) naturally induced by the boundaries. Each hypercube can be labelled by an n-tuple $\tilde{x} = (\tilde{x}^1, \ldots, \tilde{x}^n)$ with integers, $\tilde{x}^i$ with $1 \leq \tilde{x}^i \leq \tilde{n}_i$ for each i, such that this hypercube is defined as the bounded region in $\mathbb{R}^n$

$$H_x(\tilde{x}) = \{x \in \mathbb{R}^n | \beta_{\tilde{x}^i - 1}^i \leq x^i \leq \beta_{\tilde{x}^i}^i\}.$$

It can be seen that the set of boundaries (2) define $p = \prod_i \tilde{n}_i$ hypercubes. We can identify a discrete state by the n-tuple $\tilde{x}$, an integer $\tilde{x} \in \{1, ..., p\}$, or a boolean vector $\hat{x} \in \{0, 1\}^p$ such that $\hat{x} = [0, 0, ..., 0, 1, 0, ...0]^T$ and the 1 is at the $\tilde{x}$-th position.

Two discrete states $\tilde{x}_1$ and $\tilde{x}_2$ are said to be adjacent if their corresponding hypercubes $H_x(\tilde{x}_1)$ and $H_x(\tilde{x}_2)$ share an $n-1$ dimensional boundary $\mathcal{B}_{\tilde{x}_1, \tilde{x}_2}$. The transition from one discrete state to another is called a *discrete event* and is denoted $\tilde{x}_1 \rightarrow \tilde{x}_2$.

Discrete inputs: In accordance with the state-space, also the input-space can be partitioned into hypercubes. Each hypercube $H_u(\tilde{u})$ induced by the boundaries in the input-space is labelled by an m-tuple $\tilde{u} = (\tilde{u}^1, \ldots, \tilde{u}^m)$ with integers, $\tilde{u}^i$ with $1 \leq \tilde{u}^i \leq \tilde{m}_i$ for each i. In this case the set of boundaries define $q = \prod_i \tilde{m}_i$ hypercubes.

Transition function: The idea is to examine the flow generated by the dynamical system (1) along the boundary $\mathcal{B}_{\tilde{x}_1, \tilde{x}_2}$ separating the two regions that define the discrete states. In order to asses if a transition between two adjacent states is possible or not, we need to look at the sign of a coordinate function of f on the separating boundary. Concretely, if we want to decide wether a transition is possible from $\tilde{x}_1$ to $\tilde{x}_2$, we shall start by checking first the extremal points of $\mathcal{B}_{\tilde{x}_1, \tilde{x}_2}$. These are the points of coordinates $x^i = \beta_{m_1^i - 1}^i$ or $\beta_{m_1^i}^i$, $i \neq j$, and $x^j = \beta_{m_1^j}^j$. If the value of f_j in any of these points is positive, we conclude by continuity that there is also a point in the interior of $\mathcal{B}_{\tilde{x}_1, \tilde{x}_2}$ in which f_j takes a positive value and the transition from $\tilde{x}_1$ to $\tilde{x}_2$ is possible. In most cases the values of f_j in all these points (there are 2^{n-1} of them) are negative, so it is necessary to search for an eventual positive value of f_j inside $\mathcal{B}_{\tilde{x}_1, \tilde{x}_2}$. This can conveniently be done using an optimization procedure, which searches the maximum of f_j on $\mathcal{B}_{\tilde{x}_1, \tilde{x}_2}$. If the maximum value is also negative, then we can conclude that no transition from $\tilde{x}_1$ to $\tilde{x}_2$ is possible. If the maximum value is positive, then we may conclude that the transition is possible.

Computational effort

From the previous it follows that constructing a discrete-event model from a continuous systems requires a possibly large number of optimizations. It is straightforward to compute the number of optimizations NO that is necessary. For each discrete input (i.e. $\prod_{k=1}^m \tilde{m}_k$ times) the following computations have to be performed for each coordinate (i.e. n times); for the i-th coordinate, $\tilde{n}_i - 1$ boundary planes have to be checked and for each such boundary plane there are $\prod_{j \neq i} \tilde{n}_j$ boundaries $\mathcal{B}_{\tilde{x}_1, \tilde{x}_2}$ separating two states. Generally, for each such boundary an optimization has to be performed. With this, the total number of optimizations NO becomes:

$$NO = \sum_{i=1}^{n} ((\tilde{n}_i - 1)(\prod_{j \neq i} \tilde{n}_j)(\prod_{k=1}^{m} \tilde{m}_k))$$

Furthermore, in the most general case when also the input space is continuous, each optimization is performed over $m(n-1)$ variables.

Hierarchical structure

To reduce the number of optimizations the sparsity of the differential equation (1) can be exploited. Partition the state-space in ν sub-spaces, such that $R^n = R^{n_1} \times ... \times R^{n_\nu}$. The new state z is written as $z = [z_1, ..., z_\nu]^T$, where $z_i^j \in X_i \subset \{x^1, ..., x^n\}$ for $j = 1, ..., n_i$, and $i = 1, ..., \nu$. Furthermore $X_p \cap X_q = 0$ for $p \neq q$. Partition the differential equations in (1) accordingly such that we have ν sub-systems,

$$\dot{z}_i = f_i(x, u)$$

Note, that each sub-system i may only be affected by a subset of the coordinates of the original state-vector x and the input-vector u.

The computational effort to obtain discrete-event models of all these sub-systems may be significantly less than creating the complete model at once. The information provided by these sub-models then can be used to reconstruct the complete discrete-event model that would result from the original procedure.

By creating a hierarchical structure, it is possible to use the sub-models explicitly instead of building one large model. In this case, a supervisor is used to extract the necessary information for each sub-model and to reconstruct the complete state from the information provided by the sub-models. For this, the discrete state $\tilde{x} = (\tilde{x}^1, ..., \tilde{x}^n)$ and the discrete input $\tilde{u} = (\tilde{u}^1, ..., \tilde{u}^m)$ are split into parts $\tilde{x}_i, \tilde{u}_i$ consisting of a subset of the coordinates of the discrete state $\tilde{x}$ and the discrete input $\tilde{u}$. These parts $\tilde{x}_i, \tilde{u}_i$ provide the information that is necessary for sub-model i to compute (in parallel with the other models) the next possible positions of the coordinates $\tilde{x}_i^j$ of the discrete state $\tilde{z}_i$, used to construct $\tilde{x}$. This is depicted in Fig. 1.

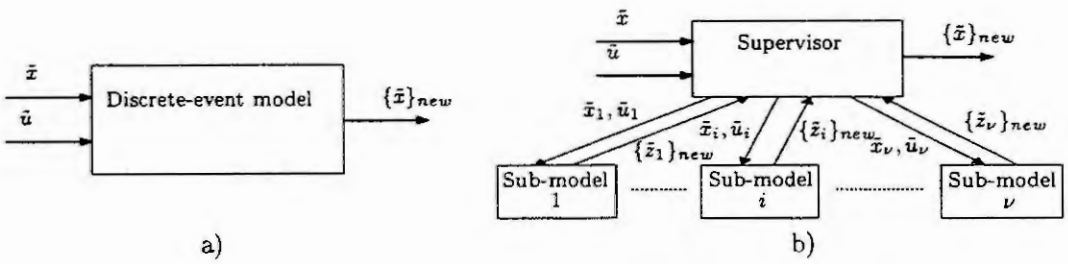


Figure 1: The a) complete- and b) hierarchical discrete-event model

Next, define the sets

$$S_{x_i} = \{j \mid \frac{\partial f_i(x,u)}{\partial x^j} \neq 0\}, \text{ and } S_{u_i} = \{j \mid \frac{\partial f_i(x,u)}{\partial u^j} \neq 0\}$$

For each sub-model i the computational effort to produce the discrete-event model is

$$NO_i = \sum_{j \in \{p \mid x_i^p \in X_i\}} ((\tilde{n}_j - 1)(\prod_{k \in S_{x_i} \setminus j} \tilde{n}_k)(\prod_{l \in S_{u_i}} \tilde{m}_l))$$

involving $\#(S_{u_i})(\#(S_{x_i}) - 1)$ variables, with $\#(S)$ denoting the cardinality of S.

State reduction

Besides the reduction of the computational effort, another advantage of using these sub-models is that it is easier to reduce the 'size' of an individual sub-model. Since a sub-model contains less information (only for some coordinates) than the complete model, it is more likely that the same information can be represented by a smaller model, i.e. a model with less states. Naturally, these smaller models must contain the same information as the original models, which can be checked looking at a simple condition. For the latter, it is convenient to represent the transition function by means of a transition (adjacency) matrix. For the complete system, a transition matrix $\hat{A} \in \{0,1\}^{p \times p}$ is a boolean matrix where $a_{ij} = 1$ if $i = \phi(j, u)$ and $a_{ij} = 0$ else. Here, i, j are integer representations of the discrete states. If the discrete

state is also represented as a boolean vector, i.e. $\hat{x} \in \{0,1\}^p$, then it is easy to compute the new set of states, given an initial state, that is $\hat{x}_2 = A\hat{x}_1$. Suppose we are given a transition matrix $\hat{A}$ for some system. We want to reduce the number of discrete states for this system by agglomerating states. This is done by the operation $\hat{x}^* = M\hat{x}$, where $\hat{x}^*$ is the new discrete state ($\dim(\hat{x}^*) < \dim(\hat{x})$) and M is a boolean matrix with $m_{ij} = 1$ and $m_{ik} = 1$ if state j and k merge and become state i. The new model can be computed readily: $\hat{A}^* = M\hat{A}M^T \& \bar{I}$, where $C = A\&B$ is defined as $c_{ij} = a_{ij}\&b_{ij}$ and $\bar{I}$ is a matrix of ones, except for the main diagonal which consists of zeros. It is easy to check that no information is lost when $\hat{A}^*M = M\hat{A}$ holds. This condition is more likely to be satisfied for the small sub-models then for the complex plant.

Example

Consider the three tank system depicted in Fig. 2.

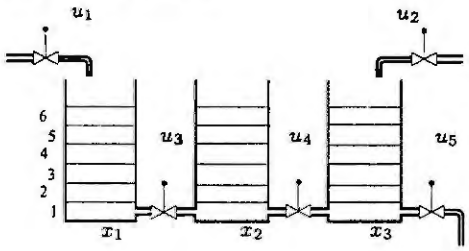


Figure 2: The three tank system

The system has three states ($n = 3$) and each state has seven boundaries ($\bar{n}_i = 6$). There are five inputs ($m = 5$) and each input has two values; 'on' and 'off' ($\bar{m}_i = 2$). To compute the discrete event-model of this system we need $\Sigma_{i=1}^{3}(6-1) * (6*6) * 2^5 = 12480$ optimizations. If we compute the discrete-event models of the three tanks separately then for the first tank we need $(6-1) * 6 * 2^2 = 120$ optimizations. For the second tank we need $(6-1) * 6 * 6 * 2^2 = 720$ optimizations. Finally, for the third tank we need $(6-1) * 6 * 2^3 = 240$ optimizations. With this, in total, we need 1080 optimizations instead of the 12480 we needed originally.

Conclusion

The sparsity of the differential equation describing a system is exploited for reducing the number of computations necessary to obtain a discrete-event model of the system. For this, the system is divided into sub-models. By using a hierarchical structure, it is possible to use the sub-models in parallel instead of building one large model. Reducing the number of states for a sub-model is more likely to be successful than for the overall system. Our example yields a reduction from 12480 optimizations to 1080.

References

[1] J. Lunze. Qualitative modelling of linear dynamical systems with quantized state measurements. *Automatica*, 30(3):417–431, 1994.

[2] Patrick Philips, Udo Bruinsma, Martin Weiss, and Heinz A. Preisig. A mathematical approach to discrete-event dynamic modelling of hybrid systems. In *Proceedings IFAC Symposium on AI in Real-Time Control*, Kuala Lumpur, Malaysia, September 1997.

[3] Heinz A. Preisig. A mathematical approach to discrete-event dynamic modelling of hybrid systems. *Computers & Chemical Engeneering*, 20:S1301–S1306, 1996.

[4] Eduardo D. Sontag. *Mathematical Control Theory: Deterministic Finite Dimensional Systems*, volume 6 of *Texts in Applied Mathematics*. Springer-Verlag, New York, 1990.

[5] J. A. Stiver and Panos J. Antsaklis. Extracting discrete event system models from hybrid control systems. In *Proceedings of the 1993 International Symposium on Intelligent Control*, pages 298–301, Chicago, Illinois, USA, August 1993.

USING WAVELETS FOR THE DETECTION OF DISCRETE EVENTS IN TIME SERIES OF HYBRID SYSTEMS

Silke Simon and Sebastian Engell
Process Control Laboratory, Department of Chemical Engineering
University of Dortmund, D-44221 Dortmund, Germany
Email: s.simon@ct.uni-dortmund.de

Abstract. This contribution deals with the use of wavelets for the analysis of time series of systems which are hybrid in the sense that they contain discrete and continuous dynamics. We focus on the detection of discrete events which is an important step in the identification of hybrid systems. A brief overview of the characteristics of the wavelet transform is given, which shows that the wavelet transform is an appropriate method for the analysis of time series of hybrid systems. By the combination of two wavelet-based analysis techniques, a two-step procedure is obtained which allows the detection of switching points in the presence of weak noise. The procedure and the influence of its parameters are demonstrated for a time series obtained from the simulation of a nonlinear laboratory plant.

Introduction

The behaviour of most technical processes is generated from the interaction of discrete and continuous subsystems: Within certain regions of the state space the system evolves continuously. When the boundaries of these regions are reached, the continuous dynamics change discontinuously due to either discrete control actions or physical phenomena. This enforces jumps in derivatives of the continuous state variables. During the last years, a lot of effort has been spent on developing *theoretical* models for such *hybrid* systems that can be used for simulation or verification of discrete controllers [1][5]. However, if the underlying physics are not well known, an *identification* based on measured data is necessary which requires, as a first step, the detection of the unknown switching points [6].

On the other hand *wavelet* theory has emerged as a powerful framework for the analysis of signals that exhibit phenomena on different time scales [3]. By the projection of a signal on wavelets, a multiresolution representation is obtained which provides information about the contributions of different frequencies at distinct times. This contribution illustrates, that wavelet theory is also a convenient framework for the analysis of time series generated by hybrid systems.

The following section explains some important characteristics of the wavelet transform and applications which make use of these properties. Subsequently, the example of a system of two coupled tanks is described. Time series of this system are used in the sequel to demonstrate how the multiresolution representation can be exploited for the detection of discrete events in time series of hybrid systems.

Wavelets: Theory and Applications

This section gives a short introduction into wavelet theory (see [3] for further information) and points out some characteristics of the transform. The basic idea of the *wavelet transform* is the decomposition of a signal y into a set of elementary building blocks

$$\psi_{a,b}(t) := \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right), \quad a, b \in \mathbf{R}, a > 0. \tag{1}$$

These are derived by scaling and translation operations from a so-called *mother wavelet* ψ, which is localised in time and frequency. Consequently, the coefficients of the transform

$$(T_\psi y)(a,b) := \int_{\mathbf{R}} y(t) \cdot \psi_{a,b}(t) dt \tag{2}$$

give not only information about the frequencies which are contained in a signal $y(t)$ but also about how the frequency content varies over time. Moreover, since for a decreasing scaling parameter a the wavelet 'zooms in' on a shorter and shorter time interval, the temporal localisation gets finer for higher frequencies.

In practice, the scale parameter usually is restricted to discrete values $a = 2^j, j \in \mathbf{Z}$ only. Moreover, if the signal is given by values at discrete time instances $y = \{y_k, k \in \mathbf{Z}\}$ the translation parameter b at each discrete scale

can be chosen to be equal to the discrete sampling instances $b = k$. By the projection of y on the resulting finite set of basis functions $\psi_{j,k}(t) = 2^{-j/2}\psi(2^{-j}(t-k))$ the coefficients of the so-called *translation invariant* [8] or *dyadic* wavelet transform are obtained, which are denoted by $(T_\psi y)(j,k)$ in the sequel.

Besides the *representation* of a signal by its wavelet coefficients one is of course also interested in reconstructing the signal from the coefficients. An important feature to characterise how many coefficients are required to yield a reconstruction close to the original signal, is the number of vanishing moments, where the k^{th} moment is defined as $\int t^k \psi(t)\, dt$: If the transformed signal is smooth, the error of approximation and thus the decay of the amplitude of the coefficients for a decreasing scaling parameter $a \to 0$ depends only on the number of vanishing moments of the analysing wavelet. Otherwise singularities of the signal or one of its derivatives limit the local decrease of the coefficients.

Mallat and Hwang [7] exploit this fact for the detection and characterisation of singularities: They show that a singularity in t_0 produces a series of maxima in the absolute value of the wavelet transform the position of which converges towards t_0 with decreasing scale. Singularities can thus be located by tracing the maxima from coarser to finer scales. In the same way, singularities of the N-th derivative can be detected, provided the analysing wavelet ψ has at least N vanishing moments. Furthermore, the regularity at t_0 can be estimated from the evolution of the coefficients along the corresponding maxima. For the characterisation of the local regularity Mallat and Hwang use *Lipschitz exponents* which they define as the supremum of all α for which two constants C and $h > 0$ and a polynomial P_n of degree $n < \alpha$ exist, so that $|y - P_n(t-t_0)| \le C|t-t_0|^\alpha$ for $|t-t_0| < h$. Using this definition, they show that y is Lipschitz α in t_0 if there exists a constant A such that along the corresponding maxima line

$$2^{-j/2} \cdot \log_2|(T_\psi y)(j,k)| \le A + \alpha \cdot j, \qquad (3)$$

subject to the condition that a wavelet with at least $N > \alpha$-1 vanishing moments has been used.

Moreover, the fact that singularities affect the decrease of the coefficients only locally enables sparse representations of signals with sparse singularities: In regions where the signal is smooth a close approximation can be obtained by reconstructing the signal using only coefficients of coarser scales. However, close to local singularities only few coefficients of finer scales are needed to yield a better approximation (*local refinement*). This property forms the basis of the so-called *Wavelet Shrinkage* [2][4], that can be used for signal denoising.

If a noisy signal with sparse singularities is transformed, the original signal can be compressed into few coefficients while noise affects all coefficients similarly. By setting most of them to zero the noise can thus be eliminated. Usual denoising schemes typically neglect the highest frequency bands for which most of the coefficients are dominated by noise. The multiresolution representation however enables to chose those coefficients *within* each frequency band that contribute significantly to the signal. By using only these coefficients for the reconstruction, the noise can be reduced and at the same time local high-frequency phenomena are retained.

The selection of significant coefficients usually is performed by thresholding: Only those coefficients, which exceed a predefined threshold λ are selected and used for the signal reconstruction. Donoho and Johnstone [4] propose the use of following '*universal*' threshold

$$\lambda = \sqrt{2 \cdot \log n} \cdot \sigma \qquad \text{(with } n \text{ the length of the signal to be denoised).} \qquad (4)$$

to yield a noise-free reconstruction. If orthonormal wavelets are used, the variance σ of the superimposed noise can be estimated from the finest scale coefficients of the transform.

Example: The Two Tanks System

The characteristics of hybrid systems mentioned in the introduction as well as the procedure to detect discrete events are demonstrated by means of the following example of a small laboratory plant (see figure 1).

It consists of two cylindric tanks which are situated at different levels. The incoming flow is controlled by a valve that switches immediately between two positions when the level in the second tank hits the upper and lower boundaries ($h_{2,max/min}$). Additionally, the system shows hybrid behaviour since depending on whether the level in the second tank exceeds the threshold H the tanks are either decoupled or coupled so that the continuous dynamics of the levels are governed by different systems of nonlinear differential equations.

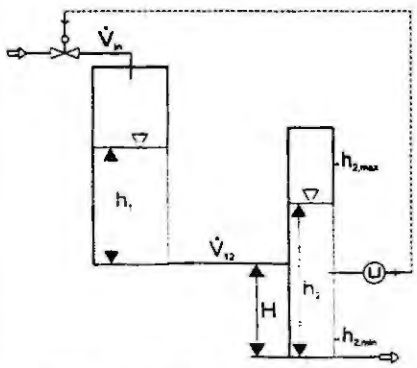


Figure 1: Scheme of the plant

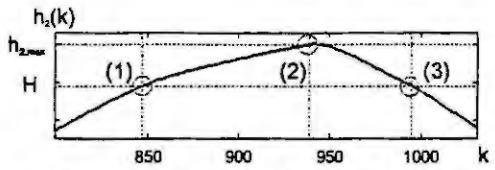


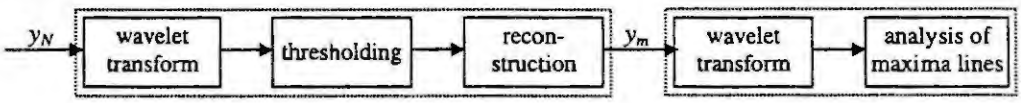
Figure 2: Time series of the two tanks system

Time series were obtained by simulation of a plant model with MATLAB-STATEFLOW. In the time series, hybrid phenomena (depicted as circles in figure 2) cannot be detected easily. An investigation of the set of differential equations shows that due to the switchings of the continuous dynamics the second derivative of the liquid level in the second tank exhibits jumps.

Wavelet Based Detection of Discrete Events

Combining the aforementioned characteristics of hybrid systems and the wavelet transform, the representation of time series of hybrid systems by their wavelet coefficients can be exploited in the following way: In regions where the system evolves continuously so that the resulting trajectory is smooth, the amplitude of the coefficients decreases fast over the scales. The time series can therefore be characterised by its coefficients on coarser scales. Discrete events however limit the local regularity (measured by the Lipschitz exponent α as described before) to the order N of the derivative in which - due to the discrete event - a discontinuity occurs. According to the results of Mallat and Hwang the switchings thus produce a series of maxima that converges with decreasing scale towards the switching point, provided the analysing wavelet has at least N vanishing moments.

In the case of no noise, the decrease of the amplitude of the wavelet coefficients over scales can be used to characterise the discontinuity (that is in which derivative a jump takes place) and to distinguish jumps caused by discrete events from inflexion points. However in the case of noise the following problem arises, if discontinuities in derivatives ($\alpha \geq 1$) are considered: Due to the faster decrease of the coefficients, the finest scales are dominated by noise and therefore cannot be evaluated for the location and characterisation as described before. Therefore the detection is combined with the wavelet shrinkage procedure to eliminate part of the noise in advance. This results in the following two step procedure:



The signal y_N is denoised first using the method of translation invariant wavelet shrinkage. The denoised time series y_m is transformed again in the second step. The resulting maxima lines are evaluated afterwards for the localisation of discrete events. Moreover, as proposed by Mallat and Hwang, maxima which correspond to discrete events can be distinguished from those of the remaining noise by considering their evolution over the scales.

Let us illustrate these steps by means of a time series of the aforementioned two tanks example, which is superposed by weak noise (signal-to-noise ratio = 200). Figure 3 shows the coefficients of the noisy time series. For this first decomposition, an orthonormal *Daubechies Wavelet* with two vanishing moments is used. As can be seen for the coarser scales, discrete events (marked by vertical lines on each discrete scale j) are 'transformed' into maxima. The corresponding coefficients are thus more likely to exceed the universal threshold and hence to be considered in the reconstruction. In this way a finer reconstruction and smaller distortion of discrete events can be achieved. However as stated before the finer scales are dominated by noise.

Figure 4 compares the coefficients of the time series which has been reconstructed from the selected coefficients (marked by circles in figure 3) to those of the noise-free time series. In this case, a spline wavelet with two vanishing moments was used because of its greater smoothness. As one can see, the proposed combination permits to 'reconstruct' partly the maxima of finer scales exploiting the redundancy of the coarser scales coefficients. Jumps in the second order derivative due to the coupling/ decoupling of the tanks as well as the switching of the incoming flow can thus be located by tracing those maxima lines which occur up to a predefined maximum scale J_{max} (here: $J_{max} = 5$). On the other hand, most of the maxima which correspond to artefacts from the denoising procedure decrease with ascending scales.

According to the aforementioned relation between the evolution of the coefficients across scales and the local regularity (Eq.3) figure 5 depicts the decrease of $2^{-j/2}\log_2|(T_\psi y)(j,k)|$ with an ascending discrete scale parameter j. Since due to the discrete sampling and the superimposed noise the asymptotic behaviour is not accessible, the local regularity is estimated from the slope between two successive scales J_{min} and $J_{min}+1$ (here: $J_{min} = 2$). In this way a discontinuity in the second order derivative is obtained for each discrete event.

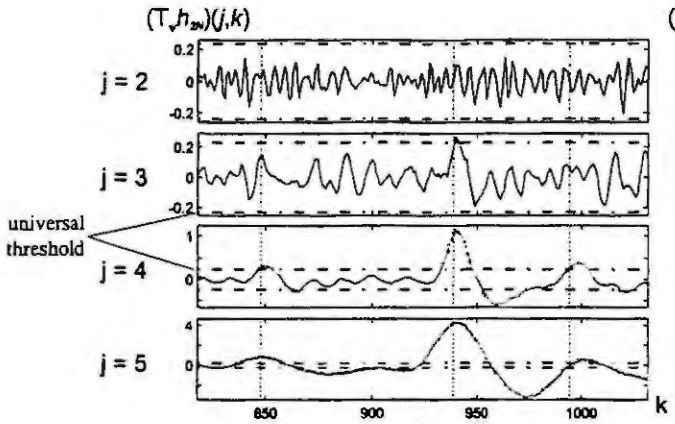


Figure 3: Thresholding the coefficients of the noisy time series

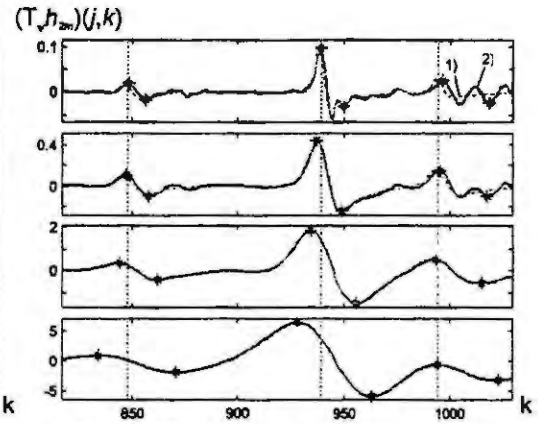


Figure 4: Coefficients of the noise-free (1) and the denoised (2) time series

If a wavelet with more than two vanishing moments is used, the coefficients of the smooth parts of the signal decay faster and thus are closer to zero for those scales which can be evaluated in practice. The local regularity can therefore be estimated more precisely. However due the larger support of wavelets with more vanishing moments, the number of coefficients which are influenced by a discrete event increases (the cone of influence associated with each discrete event expands) so that for coarser scales different singularities are more likely to affect each other. The corresponding coefficients may thus not be evaluated to characterise the local regularity.

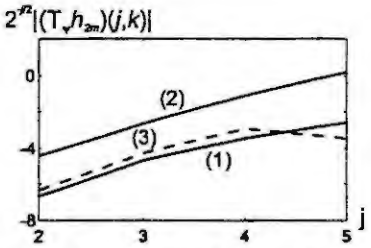


Figure 5: Estimating the order of a mode change from the decay of the coefficients across scales

In the denoising procedure, the use of a wavelet with a smaller number of vanishing moments was found to be useful for the following reason: Since discrete events are 'transformed' into fewer coefficients with a greater amplitude, the corresponding coefficients are more likely to exceed the threshold. In this way a larger local refinement and thus a smaller distortion of discrete events can achieved.

Conclusions

We have shown that the wavelet transform is a convenient method for the analysis of time series of hybrid systems: In the absence of noise discrete events can be located and characterised as maxima in the multiresolution representation. In the presence of noise, the multiresolution representation can be used to denoise the signal according to the principle of translation invariant *wavelet shrinkage*, without distorting to much discrete events. Furthermore, the sensitivity of the detection procedure can be reduced by combining these two procedures. However, the localisation of switching events which enforce jumps in higher order derivatives is still limited to time series with low noise levels.

References

[1] *Hybrid Systems I-V*, Springer-Series, New York, 1993-1999.

[2] Coifman, R. R. and D. L. Donoho: Translation-Invariant De-Noising. In: *Wavelets and Statistics*, A. Antoniadis and G. Oppenheim, eds., Springer, New York, 1995, 125-150.

[3] Daubechies, I.: Ten Lectures on Wavelets, Society for Industrial and Applied Mathematics, Philadelphia, 1992.

[4] Donoho, D. L. and I. Johnstone: Ideal Spatial Adaptation via Wavelet Shrinkage. *Biometrika*, **81**, pp. 425-455, 1994.

[5] Engell, S.: Modelling and Analysis of Hybrid Systems. Mathematics and Computers in Simulation, **46**, 445-464, 1998.

[6] Hoffmann, I. and S. Engell: *Identification of Hybrid Systems*. Proc. American Control Conference ACC'98, Philadelphia, **2**, 711-712, 1998.

[7] Mallat, S. and W. L. Hwang: Singularity Detection and Processing with Wavelets. IEEE Trans. Inform. Theory, 38 (2), 617-643, 1992.

[8] Nason, G. P. and B. W. Silverman: The Stationary Wavelet Transform and some Statistical Applications. In: *Wavelets and Statistics*, A. Antoniadis and G. Oppenheim, eds., Springer, New York, 1995, 281-299.

A FORMAL EXPRESSION OF TIME FOR DISCRETE-EVENTS DYNAMIC SYSTEMS

Christophe Thierry ; Jean-Marc Roussel ; Jean-Jacques Lesage
Laboratoire Universitaire de Recherche en Production Automatisée / Ecole Normale Supérieure de Cachan
61, avenue du président Wilson F-94235 Cachan cedex - France
e-mail : [thierry,roussel,lesage]@lurpa.ens-cachan.fr

Abstract. The concept of time is used in a significant way in the description of the logical systems. In the behaviour description models of logical systems like those of IEC61131-3 standard, time appears in the shape of temporal operators acting on logical variables. However, these operators are not formally enough defined to permit symbolic computation or formal verification. In this article, we propose the formal definition of two new temporal operators in an extended algebra whose definition set makes it possible to represent the temporal behaviour of the inputs/outputs variable of any logical system. From these definitions, we prove a set of 14 theorems on these operators. This set of theorems enables us to increase our capabilities of symbolic computation of complex logical expressions.

Keyworks. Boolean Algebra, logical system, Time, Control oriented models

Introduction

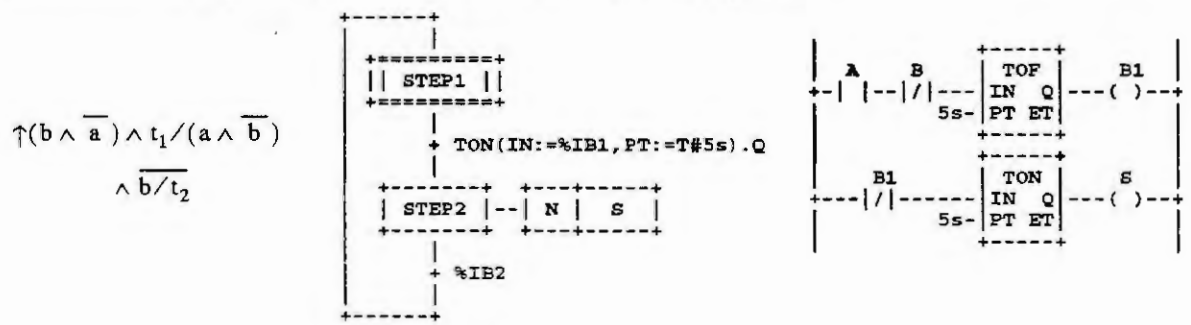
The context of the work we present in this article is the Discrete-Events Dynamic Systems modelling and analysis. We focus on description languages used in industry like Petri Nets, Sequential Function Charts, Ladder Diagram, etc... We want to present in this communication results of our work on formal definitions of primitives defined for time representation. The time description is necessary for the complete specification of any logical system. Time is found either in the parallel and sequential actions or in the definition of a delay between inputs and/or outputs. Parallelism and sequentiality are represented by the structure of the models used and the explicit time is represented in their interpretation. The time primitives allow for the representation of delay computed on input/output state changes.

This work is based on previous results presented in Mathmod 94 [4] that deal with an algebra for events modelling. Our aim is to produce a unique algebra including event and time representation in order to increase our computation capabilities.

In this paper, we will show how the notion of time is necessary in any logical systems models but also that its definition is not formal enough. We will then present the boolean algebra that is based on a representation of any logical systems signals and on the definition of basic boolean operators. The new temporal operators we introduce will be formally defined and the theorems that we proved using these operators will be presented.

Problematic

In this article, we are interested in the logical systems and more precisely in the representations of time that are included in the behavioural models associated with these systems. Whatever the logical system type, i.e. combinatorial or sequential, the concept of temporal operator is present.



logical expression — Sequential Function Charts(SFC) — Ladder Diagram

Figure 1 : *Some behavioural models for logical systems.*

Fig. 1 shows an example of logical expression, a SFC model and a Ladder Diagram model. In these three representations, a temporal operator is present (respectively b/t_2, TON(...) and a Function Block TON).

Despite the overall model behaviour may be well defined, it is to say that these temporal operators are not formally defined. Actually they are often defined using a timing diagram just as for the TON operator in the IEC 61131-3 models [2].

However, it is significant to note that these various temporal operators are all based on a single operator which is described in the electric standard IEC 617-12 [1]. This operator allows to define, starting from an input signal, an output signal whose changes of state are temporally shifted upon given values. Its definition is given in the form of the timing diagram of Fig. 2.

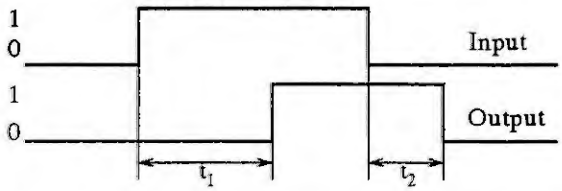


Figure 2 : *Representation of time operator* $t_1/u/t_2$ *in IEC 617-12 standard.*

Within sight of the requirements in term of simulation, evaluation or formal verification of these models [3][6][7][9], it is necessary to be able to carry out symbolic computation on expressions including these temporal operators. Thus, we propose formal definitions of two new temporal operators within an extended boolean algebra. These definitions are compatible with the definition of standard IEC617-12 and the practices of the designers of logical systems models.

The Extended Boolean Algebra II

The basic boolean algebra

The definitions we propose are based on a function set that allows the modelling of input/output of any logical system. These functions must be considered as piecewise continuous functions of time with boolean values. There are thus defined on $\mathbb{R}^{+*}$ and have their values in $\mathbb{B} = \{0, 1\}$. This set of definition have been defined as follow :

$$\mathbb{I} = \{u : \mathbb{R}^{+*} \to \mathbb{B} \mid \forall t \in \mathbb{R}^{+*} : (\exists \varepsilon_t > 0 : (\forall (\varepsilon_1, \varepsilon_2) \in]0, \varepsilon_t[^2, u(t-\varepsilon_1) = u(t-\varepsilon_2)))\}$$

This definition implies that all functions "u" of the set $\mathbb{I}$ are right continuous and allow for the existence of double discontinuity points. An example of a function "u" is given in Fig. 4. The existence of double discontinuity points ensures that $\mathbb{I}$ is closed under the operators that are to be defined.

Existent operators on $\mathbb{I}$

Using this set of definition, basic boolean operations can be formally defined as the AND, OR and NOT operators that define $\mathbb{I}$ as a boolean algebra.

In order to formalise the notion of events that is widely used in describing the dynamic behaviour of logical systems, the new operators Rising Edge (RE) and Falling Edge (FE) have been introduced in [4] and are developed in [5]. RE and FE operators allow for the definition of two new functions "$\uparrow u$" and "$\downarrow u$" that are still functions of $\mathbb{I}$.

A set of 14 theorems have been proved using these new operators so as to handle complex logical expressions including event notions. Fig. 3 presents two of the theorems proved on events operators and a expression E_1 that can be handled using these theorems.

$$E_1 = \uparrow(a \cdot \bar{c} + b) \cdot \downarrow(a \cdot b + c)$$

$$\bar{u} + \downarrow u = \bar{u}$$

$$\uparrow\left(\prod_{i=1}^{n} u_i\right) = \sum_{i=1}^{n}\left(\uparrow u_i \cdot \prod_{(j=1),(j\neq i)}^{n} u_j\right)$$

Figure 3 : *theorems on events and new computation capabilities.*

Taking time into account in this algebra

In this article, we present time operators TON (Time ON delay) and TOF (Time OFf delay) as new operators of the boolean algebra previously presented.

The TON[1] operator (Time ON delay)

This operator is formally defined with :

$$\begin{matrix} \mathbb{I} \to \mathbb{I} \\ u \to t_1/u \end{matrix} \quad \text{with } \forall t \in \mathbb{R}^{+*},\ t_1/u(t) = \begin{cases} 0 & \forall (t < t_1) \\ (\forall d \in (t - t_1, t],\ u(d) = 1) & \forall (t \geq t_1) \end{cases}$$

The TON operator transforms a function "u" into a new function "t_1/u". The state change from 0 to 1 of the function "t_1/u" is delayed from the initial function "u". For each date t, the value of this new function depends on the value of a logical expression that tests if the value of the function "u" is always true during the period t_1 of time preceding t.

In order for the new function "t_1/u" to be defined on $\mathbb{R}^{+*}$, the definition is composed of two parts. On the first part, i.e. for $t < t_1$, the period t_1 of time preceding t can not be defined. Thus, we retained that the value of "t_1/u" is false.

For the function represented on Fig. 4 for instance :

$t_1/u(d_2) = (\forall d \in (d_2 - t_1, d_2],\ u(d) = 1)$ as $d_2 > t_1$

and $(\forall d \in (d_2 - t_1, d_2],\ u(d) = 1)$ is false as $u(d_2) = 0$ and $d_2 \in (d_2 - t_1, d_2]$, i.e. $t_1/u(d_2) = 0$.

The TOF operator (Time OFf delay)

This operator is formally defined with :

$$\begin{matrix} \mathbb{I} \to \mathbb{I} \\ u \to u/t_2 \end{matrix} \quad \text{with } \forall t \in \mathbb{R}^{+*},\ u/t_2(t) = \begin{cases} (\exists d \in (0, t],\ u(d) = 1) & \forall (t < t_2) \\ (\exists d \in (t - t_2, t],\ u(d) = 1) & \forall (t \geq t_2) \end{cases}$$

The TOF operator transforms a function "u" into a new function "u/t_2". At each date t, the value of this new function depends on the value of a predicate. This predicate is true if the function "u" have been true at least once during the period t_2 of time preceding t.

As for the TON operator, the definition is divided in two parts. On the first part, i.e. for $t < t_2$, the period t_2 of time preceding t can not be defined and is replaced by a period of time from 0 to t.

For the function represented on Fig. 2 for instance :

$u/t_2(d_2) = (\exists d \in (d_2 - t_2, d_2],\ u(d) = 1)$ as $d_2 > t_2$

and $(\exists d \in (d_2 - t_2, d_2],\ u(d) = 1)$ is true as $u(d_1) = 1$ and $d_1 \in (d_2 - t_2, d_2]$, i.e. $u/t_2(d_2) = 1$.

Fig. 4 shows a function "u" of the set of definition $\mathbb{I}$ and the function "u/t_2" associated.

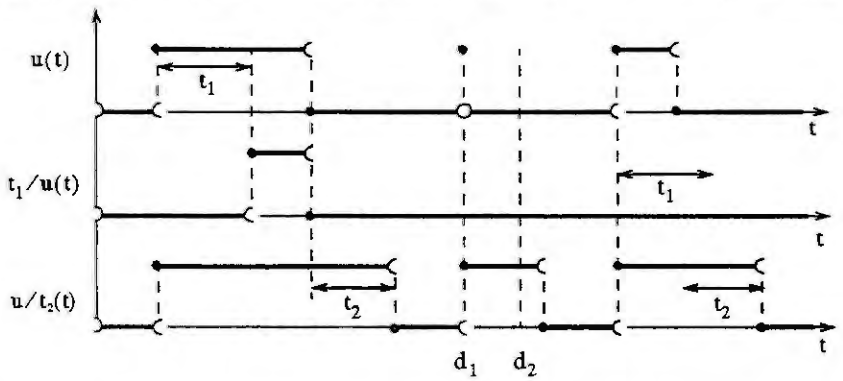


Figure 4 : *Functions "t_1/u" and "u/t_2" resulting of the TON and TOF operators applied on "u".*

By definition, $\mathbb{I}$ is closed under these two new operators as the functions "t_1/u" and "u/t_2" are defined in $\mathbb{R}^{+*}$, with boolean values and verify the property of the elements of $\mathbb{I}$ that were previously developed.

1. the TON and TOF notation are the one used in the IEC 61131-3 standard [2]

Theorems using temporal operators

Using these two new operators, we prove a set of 14 theorems. Some of these theorems are perfectly in adequacy with the practice of the DEDS models designers. All these theorems are to be used in design and analysis methods especially the ones that permit a composition of a boolean basic operator and a temporal operator.

(P1) $t_1/n + u = u$

(P2) $(u+v)/t_1 = u/t_1 + v/t_1$

(P3) $t_1/u \cdot u = t_1/u$

(P4) $t_1/(u \cdot v) = t_1/u \cdot t_1/v$

(P5) $u/t_1 + u = u/t_1$

(P6) $u/t_1 \cdot u = u$

(P7) $\overline{t_1/u}(t) = \bar{u}/t_1(t) \quad \forall t \geq t_1$

(P8) $t_1/u \cdot t_2/u = \max(t_1, t_2)/u$

(P9) $\overline{u/t_1}(t) = t_1/\bar{u}(t) \quad \forall t \geq t_1$

(P10) $t_1/u + t_2/u = \min(t_1, t_2)/u$

(P11) $u/t_1 \cdot u/t_2 = u/\min(t_1, t_2)$

(P12) $u/t_1 + u/t_2 = u/\max(t_1, t_2)$

(P13) $t_1/(t_2/u) = \mathrm{sum}(t_1, t_2)/u$

(P14) $(u/t_1)/t_2 = u/\mathrm{sum}(t_1, t_2)$

with $\mathrm{sum}(t_1, t_2) = t_1 + t_2$ (addition in $\mathbb{R}$)

Conclusions

In this paper, we have presented the result of our work on the time representation in logical systems modelling and analysis. We have shown that, although the different temporal operators are based on a unique operator, their definitions are not properly enough defined for a formal simulation or verification purpose. Thus, two new operators are presented. Their definitions is based on a formal description of input/output signal of any logical systems. Using these formal definitions, 14 theorems have been proved. These theorems increase our symbolic computation capabilities for complex logical expressions including temporal operators. We have to note that these capabilities have to be associated with the ones previously obtained on the events operators. These results, although it has an inner finality in term of formal definition of previously not well defined operators, are to be used in design, analysis and verification approaches of logical system models [8].

References

1. IEC 60617-12, Graphical symbols for diagrams - Part 12 : Binary logic elements, 1997.
2. IEC 61131-3, Programmable controllers - Programming languages, 1993.
3. De Loor, P., Zaytoon, J. and Villermain-lecolier, G., Abstractions and heuristics for the validation of grafcet controlled systems. JESA-AFCET/CNRS, Ed HERMES, Vol. 31-N°3, pp. 561-580, May 1997.
4. Denis, B., Lesage, J.-J. and Roussel, J.-M., A Boolean algebra for a formal expression of events in logical systems. In: Proc. 1.MATHMOD, pp 859-862, Vienna, 1994.
5. Lesage, J.-J., Roussel, J.-M. and Thierry, C., A theory of binary signal. In: Proceedings of CESA'96, IMACS-IEEE Multiconference on Computational Engineering in Systems Applications, pp. 590-595, Lille (France), July 1996.
6. Marcé, L., L'Her, D. and Le Parc, P., Modelling and verification of temporized Grafcet. In: Proceedings of CESA'96, IMACS-IEEE Multiconference on Computational Engineering in Systems Applications, pp. 783-788, Lille (France), July 1996.
7. Moon, I., Modelling Programmable Logic Controllers for Logic Verification. IEEE Control Systems Magazine, pp 53-59, April 1994.
8. Roussel, J.-M. and Lesage, J.-J., Validation and verification of Grafcet using state machine. In: Proceedings of CESA'96, IMACS-IEEE Multiconference on Computational Engineering in Systems Applications, pp. 758-764, Lille (France), July 1996.
9. SU, Z., Automatic Analysis of Relay Ladder Logic Programs. Report No. UCB/CSD-97-969, Computer Science Division, University of California, Berkeley, September 1997.

A DISCRETE–EVENT ABSTRACTION OF CONTINUOUS–VARIABLE SYSTEMS WITH ASYNCHRONOUS INPUTS

D. Förstner[1] and J. Lunze[2]

[1]Robert Bosch GmbH, Dept. FV/FLI
P.O.Box 10 60 50, D-70049 Stuttgart, Germany, email: Foerstner@fli.sh.bosch.de
[2]Technical University Hamburg–Harburg, Institute of Control Engineering
Eissendorfer Str. 40, D-21071 Hamburg, Germany, email: Lunze@tu-harburg.de

Abstract. The paper deals with the modelling of quantised systems, which are continuous–variable systems whose input and state variables can only be measured by a quantiser. The behaviour of the quantised system is described by event sequences. The paper proposes a purely discrete–event description of the quantised system with input events that may occur asynchronously to state events. It is shown how the transition relation of the model can be found for a given continuous system description and given event generators.

1 Introduction

Hybrid systems, in which continuous–variable and discrete–event subsystems are interconnected, pose very complex analysis, simulation, control and supervision tasks. One strategy to deal with hybrid systems is to abstract a purely discrete–event description of the whole hybrid system and to apply methods elaborated in discrete–event systems theory. This approach has been presented, for example, in [3, 6, 7, 8, 9]. The main difficulty concerns the abstraction of a purely discrete–event description of the continuous subsystem together with the injector and the quantiser (Figure 1). The result of this step can then be easily combined with a model of the discrete–event subsystem.

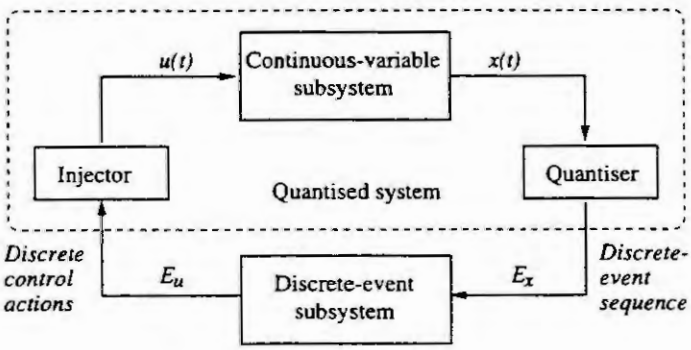


Figure 1: Hybrid system structure

Methods for abstracting discrete–event representations have either concerned discrete–time systems [3, 7] or continuous–time systems with synchronous input events [2, 4, 9]. In both situations, changes of the input $u(t)$ are assumed to occur only at the sampling instances or at the occurrence time of the output events. This simplifies the modelling task, because the input events take place at predefined time instances. This paper deals with the more general situation in which an input event may occur at any time. It is shown how the state of the discrete–event model has to be chosen and how the state transition relation of the model can be obtained for a given quantised system.

2 Quantised systems

As shown in Figure 1, the core of the quantised system is a continuous–variable continuous–time system

$$\dot{x}(t) = f(x(t), u(t)), \quad x(0) = x_0. \tag{1}$$

with $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$. The differential equation (1) is assumed to have a unique solution.

The state quantiser in Figure 1 introduces a partition of the state space $\mathbb{R}^n$ into the sets $\mathcal{Q}_x(z)$, $z \in \mathcal{N}_x = \{1, 2, \ldots, n_z\}$. The qualitative value of the state $x(t)$ at time t is given by the index z of the

set $\mathcal{Q}_x(z)$ to which the state belongs: $[\boldsymbol{x}(t)] = z \Leftrightarrow \boldsymbol{x}(t) \in \mathcal{Q}_x(z)$. The change of the qualitative value $[\boldsymbol{x}(t)]$ from i to j is called the event $e = (i, j)$. The quantised system is considered in the time interval $[0, T_h]$, where the continuous–variable system follows the trajectory $\boldsymbol{x}_{[0,T_h]}$. The quantiser generates a state event sequence $\boldsymbol{E}_x$

$$\boldsymbol{E}_x(0 \ldots H) = \text{Quant}(\boldsymbol{x}_{[0,T_h]}) = (e_0, e_1, e_2, \ldots, e_H). \tag{2}$$

$H + 1$ is the number of state events that the system generates within a given time horizon T_h. At the occurrence time t_k of the k–th event e_k the system state $\boldsymbol{x}$ assumes one value of the set $\delta\mathcal{Q}(e_k) := \delta\mathcal{Q}_x(i) \cap \delta\mathcal{Q}_x(j)$ where $\delta\mathcal{Q}_x(i)$ is the hull of $\mathcal{Q}_x(i)$. Hence, $\boldsymbol{x}(t_k) \in \delta\mathcal{Q}(e_k)$ holds.

The injector (Figure 1) associates the discrete input value $v \in \mathcal{N}_u = \{1, 2, \ldots, n_v\}$ with a quantitative value $\boldsymbol{u}^v$ such that

$$\boldsymbol{u}(t) = \boldsymbol{u}^{v_k} \quad \text{for } t_k \le t < t_{k+1} \tag{3}$$

holds for some $t_0 \ldots t_H$. The corresponding input event sequence is denoted by $\boldsymbol{E}_u = (v_0, \ldots, v_H)$.

As the sequences of state and input events are to be investigated together, the signal vector $(\boldsymbol{u}(t), \boldsymbol{x}(t))^\top$ is quantised. The resulting event sequence $\boldsymbol{E}$ describes the behaviour of the quantised system. For simplicity, it is assumed that at most one input event takes place between any two state events. Thus, $\boldsymbol{E}$ has the form

$$\boldsymbol{E}(0 \ldots 2H) = \text{Quant}\begin{pmatrix} \boldsymbol{u}_{[0,T_h]} \\ \boldsymbol{x}_{[0,T_h]} \end{pmatrix} = \begin{pmatrix} v_0 & v_1 & v_1 & v_2 & v_2 & \ldots & v_H & v_H \\ e_0 & \varepsilon & e_1 & \varepsilon & e_2 & \ldots & \varepsilon & e_H \end{pmatrix}. \tag{4}$$

The 'zero event' ε denotes that no state event occurs in the concerned step. If no input event takes place between the state events e_k and e_{k+1}, then $v_{k+1} = v_k$ holds.

3 Behaviour of the quantised system

The event sequences $\boldsymbol{E}$ are not unique [5]. That is, for a given initial event e_0 and input event sequence $\boldsymbol{E}_u$ the quantised system may generate one of a set of different event sequences and it is not possible to select the true sequence in advance. The reason for this is given by the fact that the initial state $\boldsymbol{x}_0$ of the system (1) is not exactly known but merely restricted to the set $\delta\mathcal{Q}(e_0)$. The system may produce one sequence of the set

$$\begin{aligned} \mathcal{B}_S(e_0, \boldsymbol{E}_u) \;=\; \{\, & \boldsymbol{E}(0 \ldots 2H) = \text{Quant}\left((\boldsymbol{u}_{[0,T_h]}, \boldsymbol{x}_{[0,T_h]})^\top\right) \;\mid\; \exists\, \check{t}_0, \check{t}_1 \ldots, \check{t}_{H+1} : \\ & \dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t)), \quad \boldsymbol{x}_0 \in \delta\mathcal{Q}(e_0), \quad \boldsymbol{u}(t) = \boldsymbol{u}^{v_k} \text{ for } \check{t}_k \le t < \check{t}_{k+1} \\ & \boldsymbol{E}_x = \text{Quant}(\boldsymbol{x}_{[0,T_h]}) \text{ with event times } t_0 \ldots t_H : \\ & \check{t}_0 = t_0 = 0,\ \check{t}_{H+1} = t_H = T_h,\ t_{k-1} \le \check{t}_k < t_k \text{ for } k = 1 \ldots H \;\}. \end{aligned} \tag{5}$$

Assume that a sequence of input and state events has occurred. The question whether a subsequent state event e' may occur or may not occur can be answered by a reachability analysis.

Definition 1 (Reachability set) *For given $\mathcal{X} \subset \mathbb{R}^n$ the reachability set $\mathcal{R}(\mathcal{S}, \mathcal{X}, \bar{\boldsymbol{u}})$ is defined to be the set of states $\boldsymbol{x} \in \mathcal{X}$ that a trajectory of the continuous–variable system with constant input $\bar{\boldsymbol{u}}$ may reach if the trajectory starts within the initial set $\mathcal{S} \subset \mathbb{R}^n$:*

$$\mathcal{R}(\mathcal{S}, \mathcal{X}, \bar{\boldsymbol{u}}) := \{\, \bar{\boldsymbol{x}} \in \mathcal{X} \mid \exists\, \bar{t} :\, \boldsymbol{x}(\bar{t}) = \bar{\boldsymbol{x}},\ \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \bar{\boldsymbol{u}}),\ \boldsymbol{x}(0) \in \mathcal{S},\ \boldsymbol{x}(t) \in \mathcal{S} \cup \mathcal{X} \text{ for } 0 \le t \le \bar{t} \,\}. \tag{6}$$

Hence, e and e' are two succeeding events for input $\bar{\boldsymbol{u}}$ if and only if

$$\mathcal{R}\big(\delta\mathcal{Q}(e), \delta\mathcal{Q}(e'), \bar{\boldsymbol{u}}\big) \neq \emptyset \tag{7}$$

This is illustrated by the left plot of Figure 2 for $e_0 = (8, 5)$. The initial quantitative state takes some value of the set $\mathcal{S}_0 = \delta\mathcal{Q}\,((8, 5))$ depicted by the grey area. The plot shows the corresponding trajectory bundle. The input has the constant qualitative value $v_0 = 1$ which is described by the pseudo event sequence $\boldsymbol{E}_u = (v_0, v_0) = (1, 1)$. The reachability sets for $e_1^A = (5, 4)$ and $e_1^B = (5, 2)$ are both not empty: $\mathcal{S}_1^A = \mathcal{R}(\mathcal{S}_0, \delta\mathcal{Q}(e_1^A), \boldsymbol{u}^{v_0}) \neq \emptyset$, $\mathcal{S}_1^B = \mathcal{R}(\mathcal{S}_0, \delta\mathcal{Q}(e_1^B), \boldsymbol{u}^{v_0}) \neq \emptyset$. For all other state events, the reachability sets are empty. Consequently, two events may succeed the event $e_0 = (8, 5)$, which demonstrates the nondeterminism of the qualitative behaviour $\mathcal{B}_S(e_0, \boldsymbol{E}_u)$.

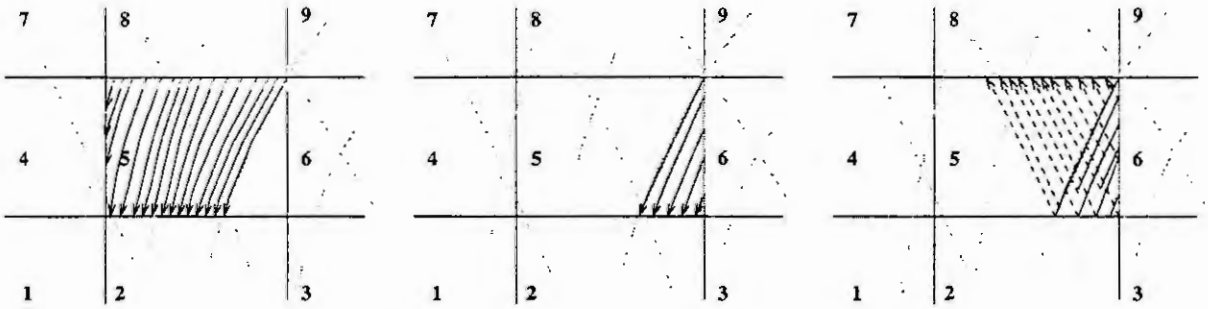


Figure 2: Behaviours for a given initial state event

Assume now that the initial event is $e_0 = (6,5)$. If an input event switches the input from $v_0 = 1$ to $v_1 = 2$ before a new state event takes place, this is an asynchronous input event. Subsequently, the state event $(5,8)$ may be generated (Figure 2, right plot). Note that the input event occurs at any time between the initial event and the next new state event. Hence, the quantised system may follow a large set of trajectories with different switching points. For this case the reachability analysis consists of two steps where the reachability sets before and after the input event are determined. If the input changes from $\bar{u} \rightarrow \bar{\bar{u}}$ before a new state event $e' = (z, z')$ takes place and if the initial event $e = (\bar{z}, z)$ brings the system into state z,

$$\mathcal{R}\Big(\mathcal{R}\big(\delta\mathcal{Q}(e),\, \mathcal{Q}_x(z),\, \bar{u}\big),\, \delta\mathcal{Q}(e'),\, \bar{\bar{u}}\Big) \neq \emptyset \;\Leftrightarrow\; e' \text{ may follow } e \text{ for switching input } \bar{u} \rightarrow \bar{\bar{u}} \qquad (8)$$

holds. The two parts of relation (8) are illustrated in the middle and right plots of Figure 2 for an initial state set $\mathcal{S}_0 = \delta\mathcal{Q}(e_0 = (6,5))$. The first plot shows the set $\mathcal{S}_1 = \mathcal{R}(\mathcal{S}_0\,,\, \mathcal{Q}_x(5)\,,\, \bar{u})$, i.e. the set of states within the actual partition $z = 5$ that may be reached before a new event occurs. Starting from $\mathcal{S}_1$, the second part of relation (8) checks whether an event e' may occur. Only for $e' = (5,8)$ the reachability set is not empty: $\mathcal{S}_2 = \mathcal{R}(\mathcal{S}_1\,,\, \delta\mathcal{Q}(e' = (5,8))\,,\, \bar{\bar{u}}) \neq \emptyset$.

4 A model for quantised systems with asynchronous inputs

A model that represents the discrete–event behaviour (5) of the quantised system has to cover the set of event sequences $\mathcal{B}_M(e_0, \boldsymbol{E}_u)$. The model has to be selected such that the relation

$$\mathcal{B}_M(e_0, \boldsymbol{E}_u) \quad \supseteq \quad \mathcal{B}_S(e_0, \boldsymbol{E}_u) \qquad (9)$$

holds. That is, the model should generate all event sequences that the quantised system may generate.

Definition 2 (Complete model) *A model that satisfies relation (9) is called complete.*

In general, the modelling aim (9) cannot be satisfied with equality sign [5]. As a consequence, the model may generate sequences that belong to $\mathcal{B}_M(e_0, \boldsymbol{E}_u)$ but not to $\mathcal{B}_S(e_0, \boldsymbol{E}_u)$. These sequences are called *spurious*. The existence of such spurious behaviours is a typical phenomenon encountered in qualitative modelling [3]. Besides the modelling aim (9) an important requirement to be satisfied when selecting a representation of the quantised system is to obtain a minimal set of spurious solutions.

A nondeterministic automaton is used to solve the representation problem. It generates the event sequences directly without a transformation from qualitative to quantitative values and vice versa. The automaton is defined by $N = (\Omega, \mathcal{V}, R, \omega_0)$ with the set of model states Ω, the set of model inputs $\mathcal{V}$, the transition relation R and the initial state ω_0. It has been demonstrated in the preceding section that state changes and input changes hold relevant information for determining the future behaviour of the quantised system. This result is exploited by the following definition of the automaton state:

$$\Omega \;=\; \{\, \omega = (e_x, e_u) \mid e_x = (z_a, z_b),\; e_u = (v_a, v_b) \,\}. \qquad (10)$$

Each state ω represents the last state event $e_x = e_x(\omega)$ and an input event $e_u = e_u(\omega)$ that occurred after the state event and before a succeeding state event. If no input event takes place, e_u assumes the 'zero event' (v_a, v_a). The set of model inputs is equal to the set of qualitative system inputs: $\mathcal{V} = \mathcal{N}_u$. The transition relation $R \subseteq \mathcal{V} \times \Omega \times \Omega$ defines which model state transitions may occur under the input v if the model is in state ω. Each transition represents an input event if e_u changes, or a state event if e_x assumes a new value. The behaviour of the nondeterministic automaton is defined as follows:

$$\mathcal{B}_M(e_0, \boldsymbol{E}_u) = \{\boldsymbol{E}(0\ldots 2H) = \begin{pmatrix} v_0 & v_1 & v_1 & \ldots & v_H \\ e_0 & \varepsilon & e_1 & \ldots & e_H \end{pmatrix} \;\Big|\; \boldsymbol{E}_u = (v_0, \ldots, v_H),\ \exists\,(\omega_0, \ldots, \omega_{2H}) :$$
$$\omega_{2k} = (e_k, (v_k, v_k)),\ \nu_{2k} = v_k \text{ for } k = 0\ldots H,$$
$$\omega_{2k+1} = (e_k, (v_k, v_{k+1})),\ \nu_{2k+1} = v_{k+1} \text{ for } k = 0\ldots H-1,$$
$$(\nu_j, \omega_j, \omega_{j+1}) \in R \text{ for } j = 0\ldots 2H-1 \quad \}. \tag{11}$$

The transition relation R of the nondeterministic automaton has to represent the discrete–event dynamics of the quantised system. The question occurs how to find this transition relation for a given quantised system such that the resulting model is complete. The key idea to solve this abstraction problem for quantised systems with asynchronous input events is to trace the reachable states represented by a model state transition. A state event e' may succeed an event e for the input $\boldsymbol{u}^v = \bar{\boldsymbol{u}}$ if a state trajectory exists from $\delta\mathcal{Q}(e)$ to $\delta\mathcal{Q}(e')$, i.e. if the reachability set (7) is not empty. Then, the tripel (v, ω, ω') is an element of the transition relation R, with $\omega = (e, (v, v))$ and $\omega = (e', (v, v))$. The crucial case of switching inputs (the input changes from $v_a \to v_b$) can be solved using eqn. (8) with $\boldsymbol{u}^{v_a} = \bar{\boldsymbol{u}}$ and $\boldsymbol{u}^{v_b} = \bar{\bar{\boldsymbol{u}}}$. If the transition may occur then $(v_b, \omega, \omega') \in R$ holds for $\omega = (e, (v_a, v_b))$ and $\omega' = (e', (v_b, v_b))$. Any input event may occur after a state event. Thus, for $\omega = (e, (v_a, v_a))$ and $\omega' = (e, (v_a, v_b))$ with $v_a \neq v_b$, $(v_a, \omega, \omega') \in R$ holds.

The reachability analysis leads to a method for determining both the discrete–event behaviour of the quantised system and the transition relation of its model. To set up the transition relation, only the last state event and input event are considered. If the full sequence of past events would be taken into account to determine possible next events, the initial set $\mathcal{S}$ would be smaller than or equal to the sets of relations (7) and (8). Consequently, the model may allow some events that can not occur in the quantised system. But it generates all possible event sequences. That is, the model is complete.

5 Conclusions

The main result of the paper is a representation of the quantised system by means of a nondeterministic automaton. The model state captures the last state event and an input event that may occur asynchronously between two succeeding state events. This allows to represent event sequences of the quantised system with input events that take place asynchronously to state events. It has been shown how the transition relation of the nondeterministic automaton can be determined by abstraction of a given quantitative system description based on a reachability analysis.

References

[1] Antsaklis, P., Kohn, W., Lemmon, M., Nerode, A., and Sastry, S. (editors), Hybrid Systems V, Lecture notes in computer science. Springer-Verlag, 1998.

[2] Förstner, D. and Lunze, J., Qualitative modelling of a power stage for diagnosis. In: Proc. 13th International Workshop on Qualitative Reasoning (QR99), Loch Awe, Scotland, 1999, pp. 105–112.

[3] Lunze, J., Qualitative modelling of linear dynamical systems with quantised state measurements. automatica, 1994, 30:417–431.

[4] Lunze, J., Process diagnosis by means of a timed discrete–event representation of continuous–variable systems. IEEE Trans. on System, Man and Cybernetics (accepted for publication), 1999.

[5] Lunze, J., Nixdorf, B., and Schröder, J., On the nondeterminism of discrete-event representations of continuous-variable systems. automatica, 1999, 35(3):395–406.

[6] Preisig, H., Pijpers, M., and Weiss, M., A discrete modelling procedure for continuous processes based on state–discretisation. In: Proc. 2nd MATHMOD, Vienna, 1997, pp. 189-194.

[7] Raisch, J., Klein, E., O'Young, S., Meder, C., and Itigin, A., Approximating automata and discrete control for continuous systems – two examples from process control. In [1], 1998.

[8] Stiver, J., Antsaklis, P., and Lemmon, M., A logical DES approach to the design of hybrid control systems. Mathl. Comput. Modelling, 1996, 23(11/12):55–76.

[9] Stursberg, O. and Kowalewski, S., Approximating switched continuous systems by rectangular automata. In: Proc. European Control Conference ECC'99, 1999.

INTEGRATED MODELLING OF RAILWAY TRAFFIC WITH PETRI NETS

Penglin Zhu and Eckehard Schnieder
Institute of Control and Automation Engineering,
Technical University of Braunschweig,
Langer Kamp 8, D-38106 Braunschweig, Germany

Abstract. This paper introduces a framework which models in an integrated way the railway system with Coloured Petri Nets (CPN). It is shown how to simplify the modelling by abstraction, modularization and reuse. The constraints upon train operations are treated centralised and exclusively in the model of the train control system so that complicated train operation algorithms can be taken into account such as resource assignment, train priority, deadlock avoidance etc.

Introduction

Railway system is a very large and highly complex system. It includes different and distributed subsystems, involves discrete/continuous, parallel/synchronised and deterministic/stochastic processes. Its modelling concerns many aspects of system modelling such as system structures, dynamic behaviours, heterogeneous system processes. The extreme system complexity is also difficult to be dealt with. It is thus a very challenge for modellers to be able to cope with such systems.

From our practices we have come to the conclusion that, in order to handle a complex system, one needs a combination of Notation, Method and Tool (BMW principle) [1]. To model a system, the modeller should at first select a suitable notation which is accompanied with a supporting tool, and then analyses and model the system with suitable methods that include for example the model structure and organisation, abstraction of details and complexity reduction, developing paradigms etc.

As modelling notations, Petri Nets possess both graphical presentations and formal mathematical foundations. Compared with other models the Petri Nets Models show following advantages: 1) ease for model understanding 2) convenience for model checking and error detection 3) possibilities of simulation and validation 4) capacity to deal with system complexity (High Level Petri Nets) 5) excellent tool support (editing, syntax checking, simulation and analysis).

The use of Petri Nets to model some subsystems and solve some problems in railway transportation can be found in the literature[2][3]. But it still seems lacking in an integrated modelling of the whole system for some investigations at system level[5], e.g. for investigation of system performances and transportation process prognosis supporting the decision-making at dispositions. One of the difficulties is probably the model complexity, because the net models may grow rapidly to a very large extent at the system modelling.

In this paper we introduce a framework which models in an integrated way the train transport processes and the functionality of the train control system. For the modelling we employ Coloured Petri Nets (CPN) [4] and the supporting tool Design/CPN [6]. One of the emphases at the modelling stage has been put on the reduction of model complexity and it shows how to simplify the modelling by abstraction, modularization and reuse. The constraints upon train operations are treated centralised and exclusively in the model of train control system so that it may take into account very complicated train operation algorithms such as resource assignment, train priority and deadlock avoidance. The Applications of the model to perform system level investigations are also briefly introduced.

System Overview

A railway system consists of the track infrastructure, the rolling stock (vehicles, locomotives etc.) and the train control system. The train operations include not only the transport processes of the rolling materials, but also the control processes in its train control and safety system.

There exist some works which model the railway systems with Petri Nets. But they often concentrate only on parts of the system, either on the transportation process [2], or on the safety technology [3] or on certain other aspects. Because no clear separation are made in these models between the transport processes and control system functions, the model will become very complex and difficult to understand when the whole system shall be modelled and investigated. In this work. we clearly separate the train movements on tracks and the system

control functions in the modelling so that different system properties can separately be handled and the modelling process be eased. The overall structure of our model are shown in Fig.1.

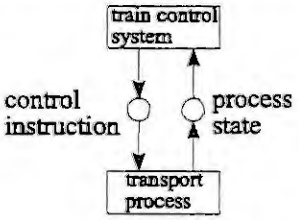


Fig.1 System structure

Train movements on tracks

In the transport processes trains move from place to place on railway tracks. The driving time of a train from its start to the goal equals to the sum of the driving times on track sections between stations and the stop times in stations under the way. That is:

$$T = \sum_i T^i_{Drive} + \sum_j T^j_{Stop}$$

If the train movement on track sections between stations and the train behaviours in stations are modelled, then the whole transport process of a train is fully described. The train movements on track sections show some repeating processes, for instance, it consists simplified of acceleration, running with constant speed and braking. The train behaviours in stations have also similar repeating properties. So it is possible and reasonable to make certain abstractions and build a class of basis models for modelling. In our model we build generic basis models for stations and track sections which can be reused and parameterized in the modelling. By combination and connection of the basic models the transport processes on a railway line can be easily modelled.

Petri Nets are composed of places, tokens in the places and transitions. To model train movements on tracks with Petri Nets, the places of Petri Nets are usually employed to represent a track section and transitions for modelling the transfers from one track section to another. The trains moving on the tracks are represented by tokens flowing through the nets.

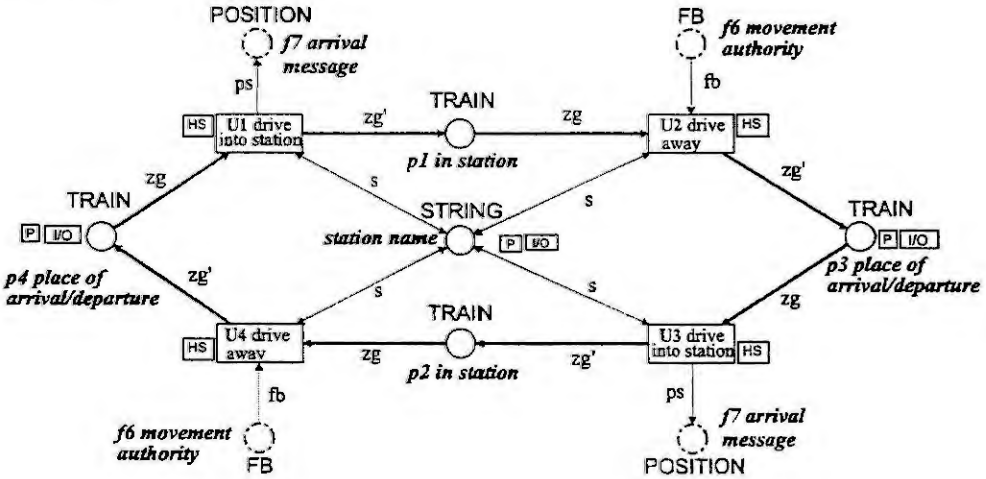


Fig.2 Station model

As an example, figure 2 shows the basis model for stations (for the meaning of notations, see [4]). The input parameters are *station name*, and train tokens on the two *arrival/departure* places at the left and right end (*p3* and *p4*). There are interface places (*f6* and *f7*) to train control system, through which the train arrival messages are sent to control system and trains get movement authorities from the control system. The thick lines indicate the train moving routes and directions. All the tracks in station are taken as the same. The substitution transitions *U1* and *U3* (consisting of subnets) model the arrival processes from the two directions left-to-right and right-to-left, in which the train stop times are determined according to the time table. The places *p1* and *p2* represent the

tracks in station and can hold many train tokens at the same time. After the stop time trains leave the station, which are modelled by the substitution transitions *U2* and *U4* for the both directions.

Modelling of train operation algorithms

Because many trains run at the same time, their movements on tracks are parallel processes which must be coordinated and regulated to avoid accidents and collisions. That is the responsibilities of the train control system. In the following we take the single track railway lines as example.

The train control system regulates the train operations with certain algorithms regarding the restrictions on resources, train type/priority, operation liveliness (free from deadlock) etc. The control algorithms are discrete and event-driven, they can be suitably modelled by Petri Nets. For the space limit we just discuss the ideas how to realise these algorithms in the modelling instead of showing the concrete models.

On a single track line, the track sections between two stations can only be occupied by trains with the same driving direction. A change of the driving direction is only allowed when there is no train in the track section. In model a train list and a direction register are assigned to each track section between two stations (Fig. 3). The tool Design/CPN [6] supports the data type *record* and *list*, so it is easy to realise this function. Before a train is allowed to enter a track section, the control system checks if a driving direction has been assigned to this section and if that is the same as the train's direction. Only when the direction of the train and the track is the same or the train list for that track section is empty, the train can drive into the track section and its train number will be added to the train list. A train will be deleted from the train list when it leaves the track section. If the train list becomes empty, then the direction of the track section is set to null.

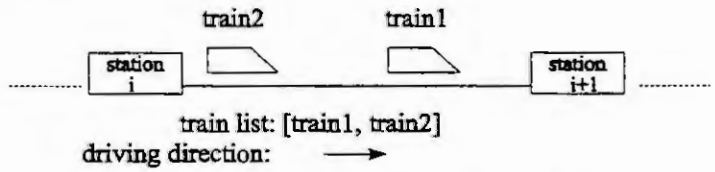


Fig.3 train list and driving direction on a track section

To take the train type/priority into consideration at the train operations, another lists are set up for each station. All trains in a station, which are ready for departure and are to run in the same direction, are put into a list for the station. When a free track section outside the station becomes available for the traffic, the train with the highest priority in the list is at first allowed to depart, then the train with next highest priority and so on.

The deadlock probability on a single track line at train operations is much greater than that on a double track line. For example, a deadlock occurs on the track in figure 4, when all the tracks in station *i* and *i+1* are occupied and all trains in station *i* shall drive to station *i+1*, and all trains in station *i+1* shall to station *i*. There are many methods to avoid the occurrence of deadlocks. Such tasks are normally taken by the dispatcher of the control system and his decisions depends strongly on his experiences. In our model the following algorithm to avoid deadlocks is modelled:

(1) Only when there is a free track in station *i*, one train from the two neighbourhood station *i+1* and *i-1* can be selected to drive to station *i*,

(2) Before selection it is examined whether all tracks in one of the two neighbouring stations are fully occupied by trains, namely the train number equals the track number ($Z_{i-1} = G_{i-1}$ or $Z_{i+1} = G_{i+1}$),

(3) If such a critical situation has been found in (2), it must be further checked if all trains in that station shall drive to station *i*. If yes, there exists a deadlock danger, a train in that station should first be allowed to drive to station *i*; if no, then no deadlock danger exists, the train selection from the two neighbouring stations can proceed on the pure basis of train priority.

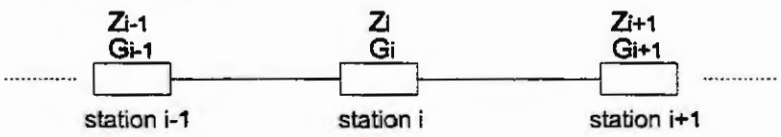


Fig.4 for the deadlock problem

Conclusions and Applications

To enable the investigation of system level performances and to provide support for the decision-making at the dispositions, the railway system has been modelled in an integrated way. We have chosen Coloured Petri Nets as the modelling language because Petri Net models possess graphic presentations and are easy for understanding, the models are executable and enable immediate simulations where error detection and model validation can be carried out conveniently.

By modelling the train transport processes and the control system functionality separately and building the abstract basic models for railway stations and track sections, the model complexity has been reduced and the modelling efficiency has been improved. The model can be used for simulating the train operation processes and establishes a basis for many system investigations. Figure 5 shows the simulated train traffic processes on the track between station 1 and station 4, given the train's arrival times on this track and the desired stop times at the stations and other parameters about the track. If the effects of stochastic influences and disturbances in railway traffic are integrated into the model, the train delays and the punctuality of the train operations can also be determined by the simulations.

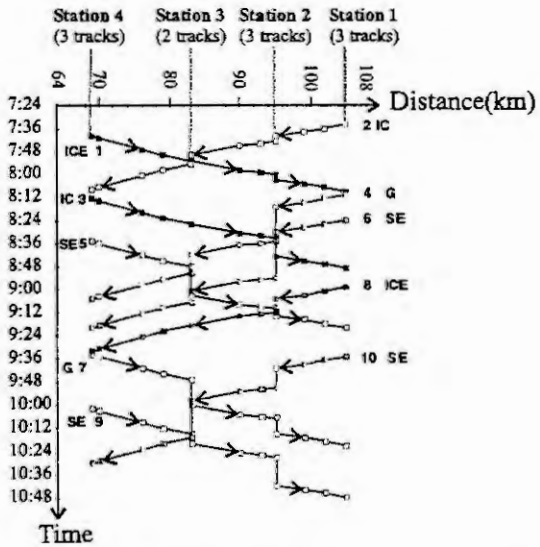


Fig.5 An example of simulated train traffic processes (time-distance diagram)

References

1. E. Schnieder, *Methoden der Automatisierung,* Vieweg, Braunschweig, 1999

2. W.M.P. van der Arlst, M.A. Odijk, *Analysis of Railway Stations by Means of Interval Timed Coloured Petri Nets,* Real-Time Systems, No. 9, 1995, p.241-263

3. M. Montigel, *Modellierung und Gewährleistung von Abhängigkeiten in Eisenbahnsicherungsanlagen,* Dissertation ETH Nr.10776, Zürich 1994

4. K. Jensen, *Coloured Petri Nets - basic Concepte, Analysis Methods and Practical Use,* volume 1, Springer-Verlag, 1992

5. P. Zhu, E. Schnieder, *Performability-Modellierung von Bahnsystemen*, In: Tagungsband des internationalen Symposiums „Eisenbahn an der Schwelle zum dritten Jahrtausend" (ZEL'99), Zilina, slowakische Republik, Mai 1999, 243-254

6. Design/CPN online, http://www.daimi.aau.dk/designCPN/

ANALYSIS AND SYNTHESIS OF HYBRID SYSTEMS USING PETRI NET-STATE-MODELS

Christian Müller, Heinrich Rake
Institute of Automatic Control, Aachen University of Technology
Steinbachstr. 54, D-52056 Aachen, Germany
Email: {mu,ra}@irt.rwth-aachen.de

Abstract. In this paper a method is presented, that is using Petri nets and switched differential equations for the modelling of hybrid systems. The so called Petri net-state-model is the basis for further investigations into discrete controlled hybrid systems with the focus on the behaviour of the discrete control. The analysis methods base on a reachability analysis of the hybrid system. For this a hybrid reachability graph, the evolution graph, is presented which is equivalent to the reachability graph for discrete event systems. In case the graphtheoretical analysis points out undesired dynamic properties of the modelled system a synthesis method is finally proposed. This method allows to restrict the occurrence of undesired state transitions by synthesising a minimum of control actions.

Introduction

The term hybrid systems has come to be used to describe systems where continuous and discrete event dynamics interact and their interaction determines the qualitative and quantitative behaviour of the system. For the continuous time part of a hybrid system this may result in changes of the continuous dynamics caused by events. Reversely, the continuous evolution may generate events by reaching some thresholds, that cause state transitions in the discrete subsystem. In the field of engineering hybrid systems are more and more of interest due to the expansion of digital devices interacting with the continuous world.

Out of the wide range of different hybrid systems [1] the focus in this paper is on those hybrid systems where continuous physical processes are controlled by a switching logic. In such systems the interactions between the often complex logic control and the coupled continuous local processes can lead to unexpected behaviour [4]. In this case a simple discrete or continuous design of the controller is usually not sufficient. The hybrid behaviour often arises from the hierarchical organisation of complex control systems where the hierarchical organisation helps manage complexity but requires increasing abstraction and the coordination of parallel processes.

Modelling, analysis and synthesis of such systems, especially of the sequential and the coordinating control, is the subject of this contribution. First a simple example is introduced for illustration. After this a modelling method for hybrid systems is presented, that consists of separate models for the continuous and discrete parts of the hybrid system, which are modular connected by integrated interfaces. In the next section analysis methods based on a hybrid reachability graph, the evolution graph, are given. Finally, a synthesis method for hybrid systems will be proposed, in case the graphtheoretical analysis points out undesired dynamic properties of the modelled system.

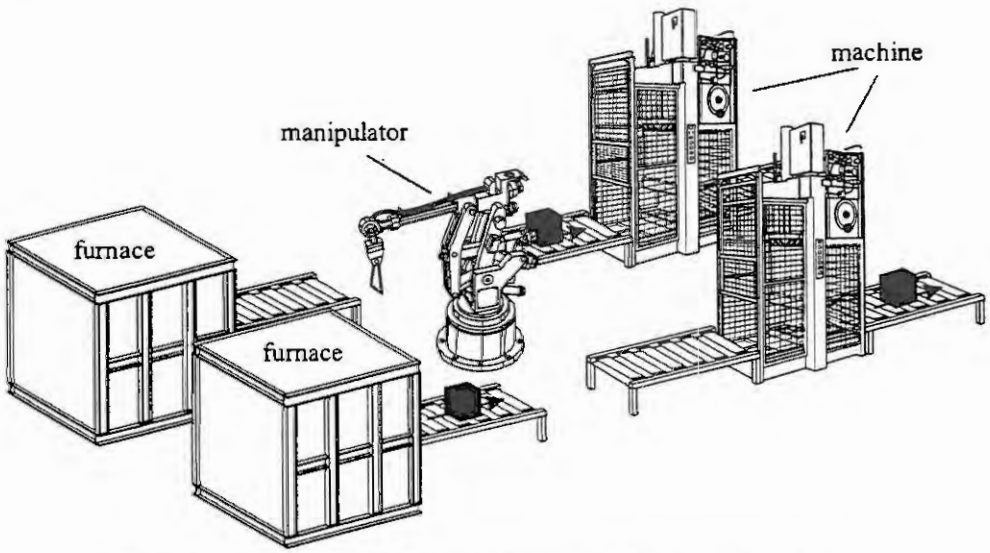


Fig. 1: Example of a manufacturing processing heat-treated work-pieces

Example

A simple example of a discrete controlled hybrid system is the manufactory in fig. 1, processing heat-treated work pieces. Two furnaces at the beginning of the process provide always enough work pieces for the processing in the two parallel working machines. There is only one manipulator in the middle (s. fig. 1) for the handling of the work pieces between furnaces and machines. The handling is identical for both production lines whereas the processing takes different times. Processed work pieces leave the manufactory immediately. The most important point is the moment a work piece leaves a furnace. Having left the furnace it has to be processed within a fixed time. Otherwise it will cool down too much, cannot be processed anymore and the process has to be stopped.

The Petri net-state-model

There are many approaches to modelling of hybrid systems [1, 5]. Is the emphasis on the design of a discrete control for a hybrid system like in this paper, it is useful to distinguish between the process on one side and the control on the other side. Thus the hybrid system is divide into its continuous and its discrete event subsystems modelled separately in the Petri net-state-model. The obtained submodels are connected by integrated interfaces via simple binary signals. Thus any continuous and any discrete submodel can be coupled with each other and the modular and efficient modelling of large systems is possible. Fig. 2 shows a simple Petri net-state-model consisting of one discrete event subsystem and a continuous one.

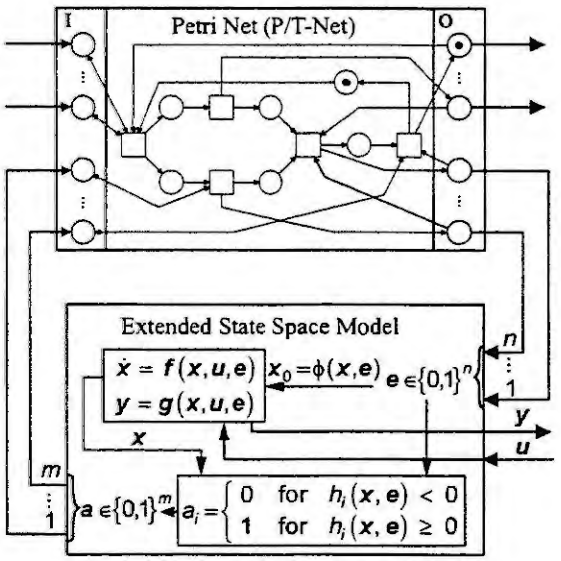


Fig. 2: Petri net-state-model

The model *DE* of a discrete event subsystem can be viewed as a triple:

$$DE = \langle N, I, O \rangle \tag{1}$$

The internal dynamic of the discrete system is modelled by a common Place/Transition-net *N*. To guarantee deterministic behaviour of the model the firing rule has to be modified. Every transition has to fire as soon as it is enabled and concurrent transitions fire in maximal steps.

To treat the binary input signals, integrated into the input vector v_I, the net is extended by the input place set $I = \{i_1, i_2, \ldots, i_n\}$ where $n = \dim(v_I)$. Thus every input signal is assigned to an input place that has, corresponding to its signal, one token (signal = 1) or is empty (signal = 0). The input places can only be connected to the net *N* by self-loops since they represent external and "read-only" firing conditions.

To create output signals, integrated into the output vector v_O, the net is extended in a way similar to the input by the output place set $O = \{o_1, o_2, \ldots, o_m\}$ where $m = \dim(v_O)$. Thus every output signal is assigned to an output place which has to be 1-safe to represent the signals 1 and 0. The output places are only additional places, thus the dynamic of the discrete event system is still determined by the net *N*.

In a model for the continuous parts of a hybrid system several hybrid phenomena like jumps or switches in the continuous trajectory and changes of the system order have to be taken into account [6]. These phenomena are integrated in the model *CD* of a continuous subsystem that can be viewed as a 7-tuple:

$$CD = \langle X, Y, U, \boldsymbol{f}, \boldsymbol{g}, \boldsymbol{h}, \Phi \rangle \tag{2}$$

Switched differential equations as an extended state space model are used to model the continuous dynamic:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x},\boldsymbol{u},\boldsymbol{e}),\ \boldsymbol{y} = \boldsymbol{g}(\boldsymbol{x},\boldsymbol{u},\boldsymbol{e}),\ \boldsymbol{x}_0 = \Phi(\boldsymbol{x},\boldsymbol{e}) \tag{3}$$

where $\boldsymbol{x}(t) \in X \subset \mathbb{R}^r$ is the dynamic state vector, $\boldsymbol{u}(t) \in U \subset \mathbb{R}^s$ the continuous input vector and $\boldsymbol{y}(t) \in Y \subset \mathbb{R}^t$ the continuous output vector. A change of the interface input signals $\boldsymbol{e} \in \{0,1\}^n$ causes a change of the system dynamics including a possibly change of the dimension of the state vector $\boldsymbol{x}$. Jumps of the state variables can be modelled by the map $\Phi(\boldsymbol{x},\boldsymbol{e})$, reinitialising the system to the initial state $\boldsymbol{x}_0$. For the communication with other, especially with discrete submodels, the binary output signals $\boldsymbol{a} \in \{0,1\}^m$ are obtained by the threshold function $\boldsymbol{h}(\boldsymbol{x},\boldsymbol{e})$.

Finally the state $X_h = \{m, \boldsymbol{x}\}$ of the hybrid system consists of the continuous state vector $\boldsymbol{x}$ and the state of the Petri net represented by the actual marking vector *m* of the net *N*.

Fig. 3 shows the hybrid model of the example. The handling and the processing are modelled in separate continuous subsystems. All continuous dynamic is reduced to its duration (t_T-handling time, t_A-max. handling time, t_{Pi}-time for processing) which leads to simple integrators in the continuous subsystems.

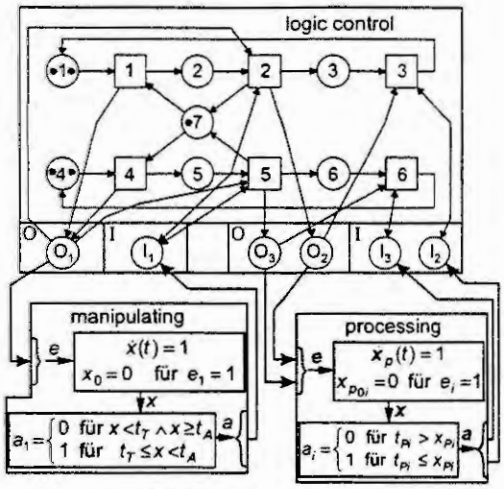


Fig. 3: Hybrid model of the example

Evolution graph and graphtheoretical analysis

Considering the dynamic behaviour of a Petri net-state-model the marking of the Petri net N changes only but then immediately when a new input signal v_i occurs. This is generated by reaching a threshold in the continuous system. For the rest of time the discrete state remain as well as the interface signals and the functions f, g and h constant. Thus a certain continuous dynamic belongs to each of this discrete states until a new threshold is reached. This period of time is called an invariant behaviour state (IB-state) [2].

All IB-states which can be reached from a given initial state X_{h0} form the hybrid reachability set $R_H(X_{h0})$. The reachability set can be represented by a directed graph, the evolution graph $E_h = \langle K, A \rangle$, where the nodes $K = R_H(X_{h0})$ correspond to the reachable IB-states and the arcs represent the transitions between the IB-states [2]. A node of the evolution graph is shown in fig. 4. Its discrete part consists of the marking of the Petri net during an IB-state and its continuous part of the initial state at the beginning of an IB-state. The arcs are labelled with the duration Δt of the IB-state, the signals a_j causing a state transition and the transitions t_j fired in the Petri net. Fig. 5 shows the evolution graph of the example.

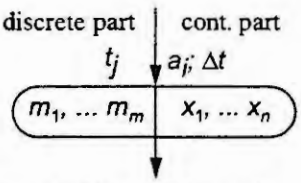


Fig. 4: Node of the evolution graph

The evolution graph is equivalent to the reachability graph for discrete event systems so that in principle graphtheoretical analysis methods of Petri nets can be used for hybrid systems, too. Because of the integration of time into the evolution graph it is also possible to consider the continuous behaviour.

Dead transitions of the Petri net, dead output signals a_j and total deadlocks of the hybrid system can be found immediately in the evolution graph. Different paths in the evolution graph arise only from classical conflicts of several transitions in the discrete event system. The evolution graph of the example (fig. 5) has no dead transitions or output signals. So all modelled processes in the manufacturing are possible. The deadlock (fig. 5 bottom left) shows that work pieces can cool down too much.

The establishment of additional properties such as lifeness, repeatability and cycles requires the condensation E_h^c of the evolution graph. This is a graph the nodes of which correspond with the strong components (set of nodes linked in both directions) of the evolution graph. A component of the condensation is life for e.g. if all transitions of the net can fire and dead if it contains only one node. Cyclic behaviour of the hybrid system is possible if there exists a component with more than one node.

The condensation of the example has eight dead components (in fig. 5 only the deadlock K_2 is marked) and the life component K_1 where several cycles are possible that realises the desired system behaviour.

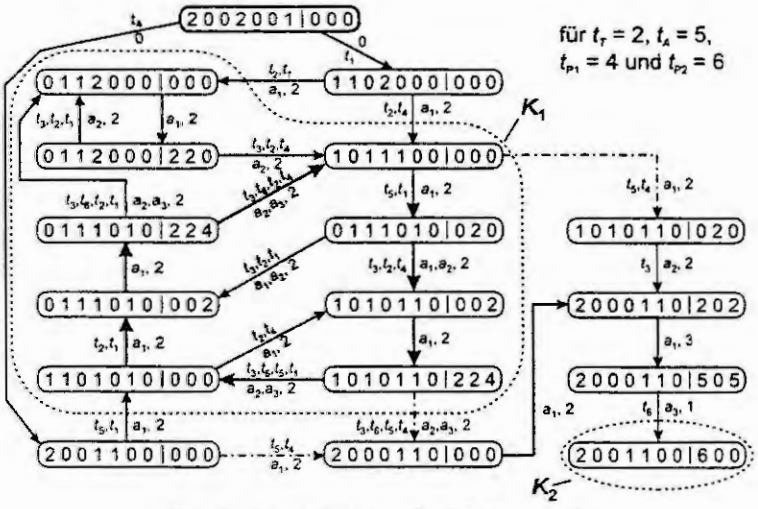


Fig. 5: Evolution graph of the example

Synthesis

After a graphtheoretical analysis of a hybrid system it is of interest how the discrete control can be modified in case the analysis points out undesired dynamic behaviour. The basic idea of the following synthesis method is to

restrict the system behaviour in order to prevent the occurrence of undesired state transitions, the critical state transitions, in the modelled system [3]. The reachability graph and its condensation representing the current dynamic properties of the hybrid system can be used directly for the correction task. In the example (fig. 5) the goal of a synthesis is to lead the process into the life condensation component K_1 and to prevent all state transitions leaving this component. Thus the process performs cyclically and no work piece cools down too much any more (component K_2). Three critical state transitions (dotted) can be found in the evolution graph of fig. 5.

The prevention can be done by disabling the critical transitions that cause the undesired state transitions. In the example there is only one critical transition, the transition t_4. Let $\boldsymbol{d}^j, \boldsymbol{a}^j = [\boldsymbol{m}^T, \boldsymbol{x}^T]^T$ denote vectors representing the hybrid states, that enable the critical transition t_j, and let $\boldsymbol{x}_h$ denote an arbitrary hybrid state vector. Suppose that the firing of t_j at a deactivator $\boldsymbol{d}^j$ is undesired whereas t_j may continue being enabled at an activator $\boldsymbol{a}^j$. Then the correction task formally can be stated as additional firing conditions

$$\boldsymbol{x}_h = \boldsymbol{d}_1^j \vee \boldsymbol{x}_h = \boldsymbol{d}_2^j \vee \ldots \Rightarrow \text{enable } t_j; \quad \boldsymbol{x}_h = \boldsymbol{a}_1^j \vee \boldsymbol{x}_h = \boldsymbol{a}_2^j \vee \ldots \Rightarrow \text{disable } t_j \tag{4}$$

for the critical transition t_j. In general it is not necessary to consider every continuous state and every place of the hybrid system to distinguish the activators and the deactivators. Only some so-called significant continuous states and places will be needed to realise a minimum of necessary control actions. To find them the matrix Δ^j is built

$$\Delta^j = [|\boldsymbol{d}_1^j - \boldsymbol{a}_1^j|; \ldots; |\boldsymbol{d}_1^j - \boldsymbol{a}_q^j|; |\boldsymbol{d}_2^j - \boldsymbol{a}_1^j|; \ldots; |\boldsymbol{d}_p^j - \boldsymbol{a}_q^j|] \tag{5}$$

by subtracting every deactivator from every activator for a critical transition t_j. A non-zero element of this matrix Δ^j means, that the corresponding place or continuous state could be used for the identification of de-/activators. In the example the critical transition t_4 has two activators and two deactivators resulting in the matrix Δ^4

$$\Delta^4 = \left[\begin{array}{cccccccc|ccc} 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 2 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 2 \end{array}\right]^T .$$

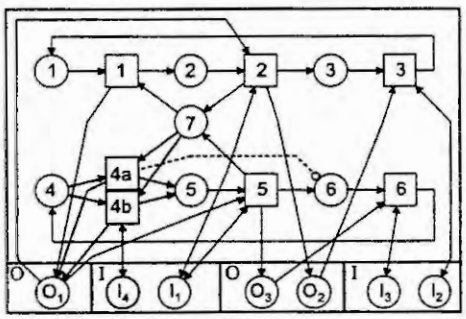


Fig. 6: Modified control of the example

Minimally one place and one continuous state, for e.g. s_6 and x_{p2} (boldly marked), are needed. This results in the additional condition $[m_6 = 0 \vee x_{p2} = 2]$ for t_4. Out of this the minimal controls can be derived that are realised in fig. 6 by self-loops and by a new threshold for x_{p2}.

Conclusion

The analysis and the synthesis presented in this contribution bases on a Petri net-state-model of the hybrid system and the evolution graph. This requires a finite evolution graph what still can't be proved. The restrictions obtained by the synthesis to prevent undesired state transitions are maximal permissive because all other state transitions may occur furthermore. The synthesis method itself is hybrid since discrete and continuous conditions normally will be arrived as shown in the example.

References

1 P. Antsaklis, X. D. Koutsoukos. On hybrid control of complex systems: a survey. in Proc. Hybrid Dynamical Systems, ADPM'98, Reims, France, pp. 1-8, March 1998.

2 R. David, H. Alla. Petri Nets for Modeling of Dynamic Systems - A Survey. Automatica, Vol. 30, No. 2, pp. 175-202, 1994.

3 W. Seiche, D. Abel. Synthesis of Deadlock-Free Control Structures Using Petri-Nets. Proc. of the IFAC Symposium on Design Methods for Control Systems, Zurich, Switzerland, 1991.

4 C. Chase, J. Serrano, P. J. Ramadge. Periodicity and Chaos from Switched Flow Systems: Contrasting Examples of Discretely Controlled Continuous Systems. IEEE Trans. Aut. Control, Vol. 38, No. 1, pp. 70-83, 1993.

5 G. Labinaz, M. M. Bayoumi, K. Rudie. Modeling and control of hybrid systems: A survey. in Proc. IFAC 13th Triennial World Congress, San Francisco, pp. 293-304, 1996.

6 M. S. Branicky, V. S. Brokar, S. K. Mitter. A unified framework for hybrid control: model and optimal control theory. IEEE Trans. Aut. Control, Vol. 43, No. 1, pp. 31-45, 1998.

UNITARY–RATE HYBRID PETRI NETS

F. Balduzzi, A. Di Febbraro*
A. Giua, C. Seatzu**

* Dip. di Automatica ed Informatica, Politecnico di Torino, Italy
email: {balduzzi,difebbraro}@polito.it.
** Dip. di Ingegneria Elettrica ed Elettronica, Università di Cagliari, Italy
email: {giua,seatzu}@diee.unica.it.

Abstract. In this paper we deal with a hybrid formalism based on Petri nets. A restricted model, called Unitary Rate Hybrid Petri Net, is defined. This model can be seen as the Petri net counterpart of a Timed Automaton. We demonstrate that the reachability problem for a hybrid net in this class can be reduced to the reachability problem of a corresponding discrete Petri net, and thus it is decidable.

1. Introduction

The control of hybrid systems, i.e., systems with both time–driven and event–driven dynamics, is a domain of increasing importance and several hybrid models have been presented in the literature.

Petri nets (PNs) [4] have originally been introduced to describe and analyze discrete event systems. Recently, much effort has been devoted to apply these models to hybrid systems as well. Among the many different hybrid net formalisms that have been proposed, we consider here a basic model that was originally presented in [2] and that was inspired from the approach of David and Alla [3]. This model, that will be called in the rest of this paper *Hybrid Petri Net* (HPN), consists of continuous places holding fluid, discrete places containing a non–negative integer number of tokens, and transitions, either discrete or continuous. Note that, unlike [2], we are assuming here that no timing structure is associated to the firing of discrete transitions.

In this paper we define a particular class of HPS called *unitary–rate HPN* (URHPN), that can be seen as the HPN counterpart of a Timed Automaton (TA) [1]. It consists of a HPN where the continuous dynamics is such that the marking of each continuous place constantly increases with a *unitary* slope. Thus the marking of each continuous place represents the value of a timer. When comparing URHPNs and TA we observe that: TA can model "reset" of the continuous state, while URHPNs can model "jumps of constant magnitude" of the continuous state (and, as in the general case, may also have an infinite discrete state space) [6].

We prove that the reachability problem is decidable for a URHPN and can be reduced to the reachability problem of a discrete PN with a suitable initial marking. This result may not be surprising, because the reachability problem is also known to be decidable for TA [1].

2. Hybrid Petri Nets

The Petri net formalism used in this paper can be seen as the "untimed" version of the model presented in [2]. For a more comprehensive introduction to place/transition Petri nets see [4].

A Hybrid Petri Net (HPN) is a structure $N = (P, T, Pre, Post, C)$.

The set of *places* $P = P_d \cup P_c$ is partitioned into a set of *discrete* places P_d (represented as circles) and a set of *continuous* places P_c (represented as double circles). The cardinality of P, P_d and P_c is denoted n, n_d and n_c.

The set of *transitions* $T = T_d \cup T_c$ is partitioned into a set of discrete transitions T_d and a set of continuous transitions T_c (represented as double boxes). The cardinality of T, T_d and T_c is denoted q, q_d and q_c.

The pre- and post-incidence functions that specify the arcs are (here $\mathbb{R}_0^+ = \mathbb{R}^+ \cup \{0\}$): $Pre: P_d \times T \to \mathbb{N}$, $Post: P_c \times T \to \mathbb{R}_0^+$.

We require (well–formed nets) that for all $t \in T_c$ and for all $p \in P_d$, $Pre(p,t) = Post(p,t)$.

The function $C: T_c \to \mathbb{R}_0^+ \times \mathbb{R}_\infty^+$ specifies the firing speeds associated to continuous transitions (here $\mathbb{R}_\infty^+ = \mathbb{R}^+ \cup \{\infty\}$). For any continuous transition $t_j \in T_c$ we let $C(t_j) = (V_j', V_j)$, with $V_j' \leq V_j$. Here V_j' represents the minimum firing speed (mfs) and V_j represents the maximum firing speed (MFS).

We denote the preset (postset) of transition t as ${}^\bullet t$ ($t^\bullet$) and its restriction to continuous or discrete places as ${}^{(d)}t = {}^\bullet t \cap P_d$ or ${}^{(c)}t = {}^\bullet t \cap P_c$. Similar notation may be used for presets and postsets of places. The incidence matrix of the net is defined as $C(p,t) = Post(p,t) - Pre(p,t)$. The restriction of C to P_X and T_Y ($X, Y \in \{c,d\}$) is denoted C_{XY}. Note that by the well-formedness hypothesis $C_{dc} = 0$.

A marking $m: P_d \to \mathbb{N}$, $P_c \to \mathbb{R}_0^+$ is a function that assigns to each discrete place a non-negative number of tokens, represented by black dots and assigns to each continuous place a fluid volume; m_p denotes the marking of place p. The value of a marking at time τ is denoted $m(\tau)$. The restriction of m to P_d and P_c are denoted with m^d and m^c, respectively. An HPN system $(N, m(\tau_0))$ is an HPN N with an initial

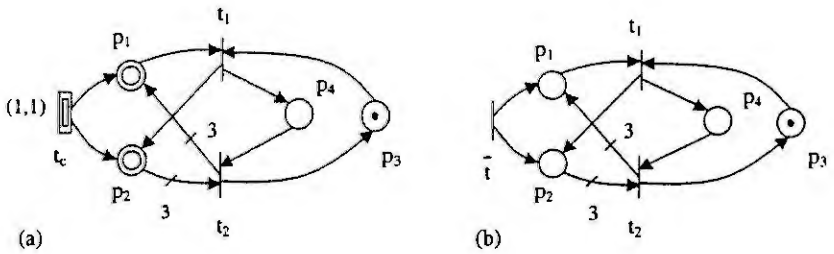


Figure 1: *(a) A URHPN; (b) the corresponding discretized PN.*

marking $m(\tau_0)$.

The enabling of a discrete transition depends on the marking of all its input places, both discrete and continuous.

Definition 1. *Let (N, m) be an HPN system. A discrete transition t is enabled at m if for all $p_i \in {}^{\bullet}t$, $m_i \geq Pre(p_i, t)$.* ■

A continuous transition is enabled only by the marking of its input discrete places. The marking of its input continuous places, however, is used to distinguish between strongly and weakly enabling.

Definition 2. *Let (N, m) be an HPN system. A continuous transition t is enabled at m if for all $p_i \in {}^{(d)}t$, $m_i \geq Pre(p_i, t)$.*

We say that an enabled transition $t \in T_c$ is: strongly enabled at m if for all places $p_i \in {}^{(c)}t$, $m_i > 0$; weakly enabled at m if for some $p_i \in {}^{(c)}t$, $m_i = 0$.

In the following we describe the hybrid dynamics of an HPN. We first consider the time–driven behavior associated to the firing of continuous transitions, and then the event–driven behavior associated to the firing of discrete transitions.

The instantaneous firing speed (IFS) at time τ of a transition $t_j \in T_c$ is denoted $v_j(\tau)$. We can write the equation which governs the evolution in time of the marking of a place $p_i \in P_c$ as

$$\dot{m}_i(\tau) = \sum_{t_j \in T_c} C(p_i, t_j) v_j(\tau) \tag{1}$$

where $\boldsymbol{v}(\tau) = [v_1(\tau), \ldots, v_{n_c}(\tau)]^T$ is the IFS vector at time τ. Indeed Equation 1 holds assuming that at time τ no discrete transition is fired and that all speeds $v_j(\tau)$ are continuous in τ.

The enabling state of a continuous transition t_j defines its admissible IFS $\boldsymbol{v}_j$. If t_j is not enabled then $v_j = 0$. If t_j is strongly enabled, then it may fire with any firing speed $v_j \in [V_j', V_j]$. If t_j is weakly enabled, then it may fire with any firing speed $v_j \in [V_j', \overline{V}_j]$, where $\overline{V}_j \leq V_j$ since t_j cannot remove more fluid from any empty input continuous place $\overline{p}$ than the quantity entered in $\overline{p}$ by other transitions.

We now characterize the set of all admissible IFS vectors.

Definition 3. (admissible IFS vectors)
Let (N, m) be an HPN system. Let $T_{\mathcal{E}}(m) \subset T_c$ ($T_{\mathcal{N}}(m) \subset T_c$) be the subset of continuous transitions enabled (not enabled) at m, and $P_{\mathcal{E}} = \{p_i \in P_c \mid m_i = 0\}$ be the subset of empty continuous places. Any admissible IFS vector $\boldsymbol{v}$ at m is a feasible solution of the following linear set:

$$\begin{cases} (a) & V_j - v_j \geq 0 & \forall t_j \in T_{\mathcal{E}}(m) \\ (b) & v_j - V_j' \geq 0 & \forall t_j \in T_{\mathcal{E}}(m) \\ (c) & v_j = 0 & \forall t_j \in T_{\mathcal{N}}(m) \\ (d) & \sum_{t_j \in T_{\mathcal{E}}} C(p, t_j) v_j \geq 0 & \forall p \in P_{\mathcal{E}}(m). \end{cases} \tag{2}$$

The set of all feasible solutions is denoted $S(N, m)$. ■

Constraints of the form (2.a), (2.b), and (2.c) follow from the firing rules of continuous transitions. Constraints of the form (2.d) follow from (1), because if a continuous place is empty then its fluid content cannot decrease.

Note that the set S is a function of the marking of the net. Thus as m changes it may vary as well. In particular it changes at the occurrence of the following macro–events: (a) a discrete transition fires, thus changing the discrete marking and enabling/disabling a continuous transition; (b) a continuous place becomes empty, thus changing the enabling state of a continuous transition from strong to weak.

Let τ_k and τ_{k+1} be the occurrence times of two consecutive macro–events of this kind; we assume that within the interval of time $[\tau_k, \tau_{k+1})$ the IFS vector is constant and we denote it $\boldsymbol{v}(\tau_k)$. Then the continuous behavior of an HPN for $\tau \in [\tau_k, \tau_{k+1})$ is described by: $\boldsymbol{m}^c(\tau) = \boldsymbol{m}^c(\tau_k) + C_{cc}\boldsymbol{v}(\tau_k)(\tau - \tau_k)$, $\boldsymbol{m}^d(\tau) = \boldsymbol{m}^d(\tau_k)$.

The firing of a discrete transition t_j at $\boldsymbol{m}(\tau)$ yields the marking: $\boldsymbol{m}^c(\tau) = \boldsymbol{m}^c(\tau^-) + C_{cd}\boldsymbol{\sigma}(\tau)$, $\boldsymbol{m}^d(\tau) = \boldsymbol{m}^d(\tau^-) + C_{dd}\boldsymbol{\sigma}(\tau)$, where $\boldsymbol{\sigma}(\tau)$ is the firing count vector associated to the firing of transition t_j.

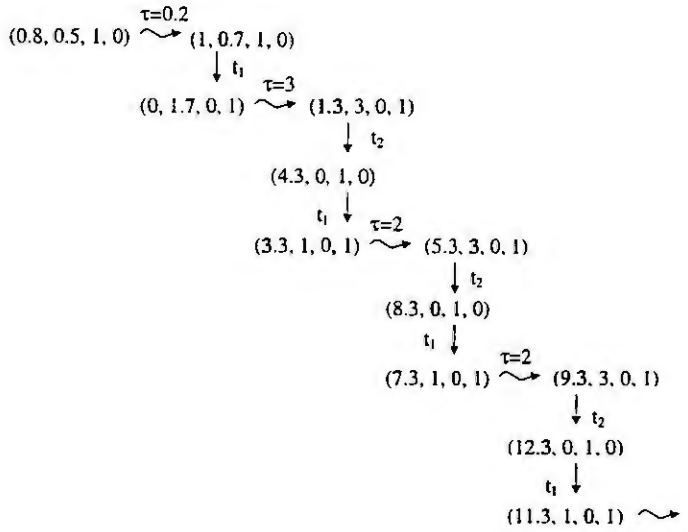


Figure 2: *The reachability graph of the URHPN in figure 1*

2.1. Firing sequence and reachability

Now, we provide some definitions that will be useful in the following.

Definition 4. *(Event Step) Let us consider a HPN system (N, m). If $t \in T_d$ is enabled at m, t may fire. The firing of t determines a new marking $\bar{m} = m + Post(\cdot, t) - Pre(\cdot, t)$ and we write $m[t\rangle\bar{m}$.* ■

We can use a similar notation for the marking variation due to the firing of continuous transitions.

Definition 5. *(Time Step) Let us consider a HPN system (N, m). If $t \in T_c$ is enabled at m for a time interval of length $\bar{\tau} \in \mathbb{R}^+$. The firing of t during that time interval determines a new marking $\bar{m}$: $\bar{m}^d = m^d$, $\bar{m}^c = \int_0^{\bar{\tau}} C_{cc} v(\tau) d\tau + m^c \geq 0$, where $v \in \mathcal{S}(N, m)$ and we write $m[\bar{\tau}\rangle\bar{m}$.* ■

Definition 6. *Let (N, m) be a HPN system. A firing sequence $\sigma = \alpha_1, \cdots, \alpha_k \in (T_d \cup \mathbb{R}^+)^*$ is enabled from a marking m if $m[\alpha_1\rangle m_1[\alpha_2\rangle m_2 \cdots [\alpha_k\rangle\bar{m}$ holds. To denote that the firing of σ from m determines the marking $\bar{m}$ we write $m[\sigma\rangle\bar{m}$.* ■

3. Unitary–rate hybrid Petri nets

In this section we define a special class of hybrid Petri nets called *unitary–rate* HPNs that can be seen as the net counterpart of timed automata.

Definition 7. *A unitary–rate hybrid Petri net (URHPN) is a HPN where: $T_c = \{t_c\}$, ${}^\bullet t_c = \emptyset$, $C(t_c) = (1, 1)$, $\forall p \in P_c : Post(p, t_c) = 1$, $Pre, Post \in \mathbb{N}^{n \times q}$.* ■

Thus a unitary–rate hybrid Petri net has a *single* continuous transition that is always enabled — because it has no input places — and whose firing speed is always unitary. The marking of all continuous places increases with unitary rate during a time step. Discontinuous variations of continuous markings may only follow the firing of discrete transitions. Furthermore, we assume that all arcs have integer weights. Such an assumption has been introduced for simplicity. In fact, whenever $Pre, Post \in \mathbb{R}^{n \times q}$ all the weights could be multiplied by the least common multiple of the denominators of all the constants appearing in $Pre, Post$ to get a new hybrid net that is isomorphic with a new one where $Pre, Post \in \mathbb{N}^{n \times q}$. Even if $Pre, Post \in \mathbb{R}^{n \times q}$ but each weight has the same irrational numbers as common factors, an isomorphism with a net where $Pre, Post \in \mathbb{N}^{n \times q}$ can be determined.

The evolution of URHPNs can be related to that of timed HA. In fact, the constant rate variation of continuous marking in URHPNs agrees with the set *Inclusions* containing the single element $1 \in \mathbb{R}^n$ in timed HA. However, all the differences outlined in the previous section still hold. In particular, in URHPNs the firing of a discrete transition may only produce constant variations on the continuous marking. On the other hand, URHPNs can assume an infinite number of discrete states.

Example 8. The HPN in figure 1.a is a URHPN. Its reachability graph is shown in figure 2 under the assumption that $m_0 = (0.8, 0.5, 1, 0)$. It has been drawn in accordance with the following rule. The firing of the continuous transition is represented only if it produces a variation on the enabling condition of the net. Note however that the continuous transition is always enabled and always fires with a constant unitary rope. Therefore, all the markings obtained from those in figure 2 with the addition of the same positive real number to m_{p_1} and m_{p_2}, are reachable. ■

Now, we prove that the reachability problem for URHPNs is decidable.

Let us first define an equivalence relation on $(\mathbb{R}_0^+)^m$.

Definition 9. *A vector $x \in (\mathbb{R}_0^+)^m$ is time-consistent with $y \in (\mathbb{R}_0^+)^m$ if: $\exists b \in [0, 1) : \forall i = 1, \cdots, m, \langle y_i \rangle = \langle x_i + b \rangle$ where $\langle \cdot \rangle$ denotes the fractional part and we write $x \sim y$. The equivalence classes of this relation are denoted $[x]$.* ■

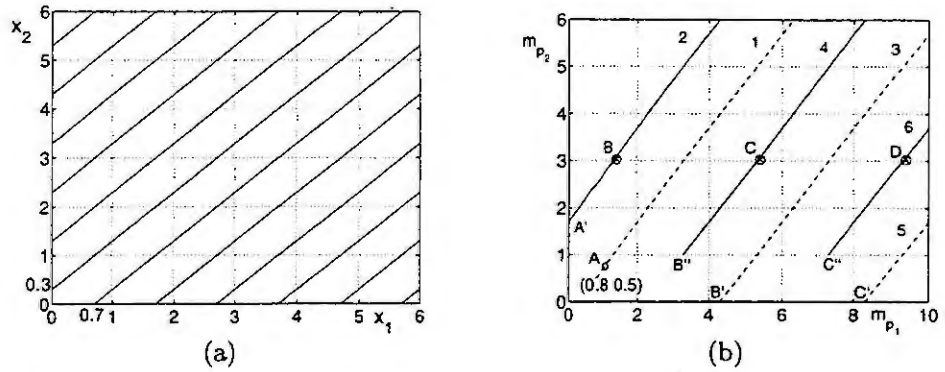


Figure 3: *(a) The equivalence class* $[(0, 0.3)]$*; (b) the set of continuous markings for the URHPN in example 12.*

Example 10. Let $\boldsymbol{x} = (0, 0.3)$. In figure 3.a the set of vectors time-consistent with $\boldsymbol{x}$ are represented in the plane (x_1, x_2) and lie on a family of parallel lines. All lines are equally spaced and are characterized by a constant unitary slope. ■

Lemma 11. *Let* $(N, \boldsymbol{m})$ *be a URHPN system. If* $\tilde{\boldsymbol{m}} \in R(N, \boldsymbol{m})$ *then* $\tilde{\boldsymbol{m}}^c \in [\boldsymbol{m}^c]$.

Proof. If $\tilde{\boldsymbol{m}} \in R(N, \boldsymbol{m})$, then there exists a firing sequence $\sigma = \alpha_1, \alpha_2, \cdots, \alpha_k$ such that $\boldsymbol{m}[\alpha_1\rangle \boldsymbol{m}_1[\alpha_2\rangle \boldsymbol{m}_2 \cdots [\alpha_k\rangle \tilde{\boldsymbol{m}}$. Since all the arc weights are integers, the firing of a discrete transition produces no variation on the fractional parts of a continuous marking. Thus, if $\boldsymbol{m}_{i-1}[\alpha_i\rangle \boldsymbol{m}_i$ and $\alpha_i \in T_d$, then $\langle \boldsymbol{m}_{i-1} \rangle = \langle \boldsymbol{m}_i \rangle$ and $\boldsymbol{m}_i^c \in [\boldsymbol{m}_{i-1}^c]$.

On the contrary, the firing of the continuous transition produces a variation on the fractional parts of the continuous marking. However, all these variations have the same magnitude. Thus, if $\alpha_i = \overline{\tau} \in \mathbb{R}^+$, then $\boldsymbol{m}_i^c = \int_0^{\overline{\tau}} \boldsymbol{C}_{cc} \boldsymbol{v}(\tau) d\tau + \boldsymbol{m}_{i-1}^c$. However, $\boldsymbol{v}(\tau) = 1$ and $\boldsymbol{C}_{cc} = 1$ by hypothesis, hence $\boldsymbol{m}_i^c = \boldsymbol{m}_{i-1}^c + \overline{\tau}$ where $\overline{\tau}$ is a vector $\in \mathbb{R}^{n_c}$ whose components are all equal to $\overline{\tau}$. Now, let $b = \langle \overline{\tau} \rangle$, then $\forall p \in P_c, \langle \boldsymbol{m}_{i,p} \rangle = \langle \boldsymbol{m}_{i-1,p} + b \rangle$. Thus, $\boldsymbol{m}_i^c \in [\boldsymbol{m}_{i-1}^c]$.

Finally, we can conclude that $\tilde{\boldsymbol{m}}^c \in [\boldsymbol{m}^c]$ by the transitivity of equivalence relations. □

Example 12. Let us consider the URHPN system $(N, \boldsymbol{m}_0)$ in example 8 with initial marking $\boldsymbol{m}_0 = (0.8, 0.5, 1, 0)$. In figure 3.b the set of all continuous markings reachable from $\boldsymbol{m}_0$ is represented. Obviously, this is a subset of $[\boldsymbol{m}_0^c]$.

Lines have been partitioned in two different sets and distinguished as dash and continuous lines. Dash lines belong to the set of continuous markings reachable when the discrete marking is equal to $\boldsymbol{m}^d = (1, 0)$, while continuous lines belong to the set of continuous markings reachable in the case of $\boldsymbol{m}^d = (0, 1)$. The discrete marking changes every times one of the discrete transition fires and discrete transitions can only fire alternatively.

Let us examine all possible evolutions of the net when the initial marking is $\boldsymbol{m}_0$. During the first 0.2 time instants, no discrete transition is enabled and t_c fires until the marking moving along line 1 reaches point A corresponding to $(1, 0.7, 1, 0)$. Now t_1 become enabled. Thus from point A it may fire changing the marking to point A'. Note however that t_1 is not required to fire as soon as A is reached; it may fire from any other point on line 1 greater than A thus reaching a corresponding point on line 2. For all markings on line 2 smaller than B no discrete transition is enabled and only the continuous transition fires until B is reached. Now t_2 become enabled. Thus from point B it may fire changing the marking to point B'. Note however that t_2 is not required to fire as soon as B is reached; it may fire from any other point on line 2 greater than B thus reaching a corresponding point on line 3. All markings on line 3 enable transition t_1 that may fire thus reaching a corresponding point on line 4. Everything repeats periodically as shown in figure 3.b. We also observe that the points A, A', B, etc. that characterize the net evolution correspond to the markings in the reachability graph of figure 2.

■

Now, let us define a transformation on a hybrid Petri net system.

Definition 13. *Given a HPN* $N = (P, T, Pre, Post, C)$*. We define the "discretized PN associated to* N*", the P/T net* $\lfloor N \rfloor = (P', T', Pre', Post')$ *with:* $P' = P$*, i.e.,* $\lfloor N \rfloor$ *has as many places as* N*, but they are all discrete;* $T' = T$*, i.e.,* $\lfloor N \rfloor$ *has as many transitions as* N*, but they are all discrete;* $Pre'(p, t) = \lfloor Pre(p, t) \rfloor$*;* $Post'(p, t) = \lfloor Post(p, t) \rfloor$*, where* $\lfloor \cdot \rfloor$ *denotes the integer part. We call* $\lfloor N \rfloor$ *the discretized HPN associated to* N*.* ■

Example 14. In figure 1.b the discretized PN corresponding to the HPN in figure 1.a is shown. ■

Now, we provide a necessary and sufficient condition for a marking $\boldsymbol{m}$ in a URHPN to be reachable.

Theorem 15. *Let* $(N, \boldsymbol{m}_0)$ *be a URHPN system. Then,* $\boldsymbol{m} \in R(N, \boldsymbol{m}_0)$ *iff* $\boldsymbol{m}^c \in [\boldsymbol{m}_0^c]$ *and* $\lfloor \boldsymbol{m} \rfloor \in$

$R(\lfloor N \rfloor, \tilde{m})$ *where*

$$\tilde{m}_p = \begin{cases} \lfloor m_{0,p} \rfloor + 1 & \text{if } \langle m_p \rangle < \langle m_{0,p} \rangle \\ \lfloor m_{0,p} \rfloor & \text{otherwise} \end{cases}$$

and $\lfloor N \rfloor$ *is the discretized net associated to* N.

Proof. First, let us observe that $m \in R(N, m_0)$ iff $\exists\, \sigma$ such that $m_0[\sigma\rangle m$. Since the continuous transition in (N, m_0) is always enabled, this implies that $\exists\, \sigma' = \sigma_\tau \sigma_T$ such that $m_0[\sigma'\rangle m$, where $\sigma_\tau \in \mathbb{R}_0^+$ and $\sigma_T \in T_d^*$, i.e., if m is reachable, then it may also be reached by a "normalized sequence" where a single time step occurs first, and all the event steps occur only at the end.

The firing sequence σ_τ can be written as $\sigma_\tau = \sigma'_\tau \sigma''_\tau$, where $\sigma'_\tau = \langle \sigma_\tau \rangle$, and $\sigma''_\tau = \lfloor \sigma_\tau \rfloor$. Therefore, $m_0[\sigma'_\tau\rangle m'_0[\sigma''_\tau\rangle m'[\sigma_T\rangle m$. Obviously, $\langle m'_0 \rangle = \langle m' \rangle = \langle m \rangle$.

We now observe that the difference in the fractional part between m_0 and m is due to the time step σ'_τ, that has a length less than one and yields m'_0 from m_0. Obviously, $\forall p \in P_c$, if $\langle m_p \rangle \equiv \langle m'_{0,p} \rangle \geq \langle m_{0,p} \rangle$ then $\lfloor m'_{0,p} \rfloor = \lfloor m_{0,p} \rfloor$. Otherwise, if $\langle m_p \rangle \equiv \langle m'_{0,p} \rangle < \langle m_{0,p} \rangle$, then $\lfloor m'_{0,p} \rfloor = \lfloor m_{0,p} \rfloor + 1$. Thus the integer part of m'_0 is exactly the marking $\tilde{m}$ defined in the theorem statement.

Finally we observe that because m'_0 and m have the same fractional part, then $m \in R(N, m'_0)$ if and only if $\lfloor m \rfloor \in R(\lfloor N \rfloor, \lfloor m'_0 \rfloor)$. In fact, let $\bar{t}$ be the discrete transition of $\lfloor N \rfloor$ corresponding to the continuous transition t_c of N. With the notation used above it is easy to understand that $m'_0[\sigma''_\tau \sigma_T\rangle m$ if and only if $\lfloor m'_0 \rfloor [\sigma''_T \sigma_T\rangle \lfloor m \rfloor$ where σ''_T contains the transition $\bar{t}$ an number of times equal to σ''_τ. Thus, $\lfloor N \rfloor$ simulates N firing $\bar{t}$ for each time step of length 1 occurring in N. □

Example 16. Let us consider the URHPN system (N, m_0) in example 8 with initial marking $m_0 = (0.8, 0.5, 1, 0)$. We want to determine if $m = (5, 0.7, 1, 0) \in R(N, m_0)$ by applying theorem 15.

Clearly $m^c \in [m_0^c]$ because if we take $b = 0.2$, then $\forall p \in P_c$, $\langle m_p \rangle = \langle m_{0,p} + b \rangle$. Then, if we consider the discretized PN in figure 1.b we see that $(5, 0, 1, 0) \in R(\lfloor N \rfloor, (1, 0, 1, 0))$ where, in accordance with the notation of theorem 15, $(1, 0, 1, 0) = \tilde{m}$ and $(5, 0, 1, 0) = \lfloor m \rfloor$. In fact, the firing sequence $\overline{\sigma} = \bar{t}, t_1, \bar{t}, t_2$ is such that $\tilde{m}[\overline{\sigma}\rangle \lfloor m \rfloor$. Therefore, we can conclude that even $m \in R(N, m_0)$. The same conclusion can be reached by looking at figure 3. In fact, it is easy to observe that the firing sequence $\sigma = 0.2, t_1, 1.3, t_2, 0.7$ is such that $m_0[\sigma\rangle m$. ■

By virtue of the above theorem 15, the results on the reachability of discrete Petri nets can be extended to URHPNs, thus proving the validity of the following corollary.

Corollary 17. *The reachability problem is decidable for URHPNs.*

Proof. Follows from theorem 15 and from the fact that the reachability problem is decidable for discrete PN [5].

□

4. Conclusions

In this paper we have defined a special class of Hybrid Petri Nets, called *Unitary Rate Hybrid Petri Nets*, that can be seen as the Petri net counterpart of a Timed Automaton. The reachability problem for a hybrid net in this class has been reduced to the reachability problem of a corresponding discrete Petri net, and thus it is decidable.

To study this class of nets, in one of the examples we have informally used the reachability graph analysis that has been developed for discrete nets. It may be interesting to find out if a technique based on the reachability/coverability graph may always be applied to this hybrid model and which properties can be studied with it.

It is also worth defining and exploring new restricted classes of HPNs. These structures may extend the classes of models for which important properties can be shown to be decidable and can be studied with standard tools of discrete Petri nets.

References

[1] R. Alur, D. L. Dill, *A Theory of Timed Automata*, Theoretical Computer Science, 126: 183–235, 1994.

[2] F. Balduzzi, A. Giua, G. Menga, "Hybrid Stochastic Petri Nets: Firing Speed Computation and FMS Modelling," *WODES'98, Proc. Fourth Workshop on Discrete Event Systems*, (Cagliari, Italy), pp. 432–438, Aug 1998.

[3] R. David, H. Alla, "Autonomous and Timed Continuous Petri Nets," *Advances in Petri Nets 1993*, G. Rozenberg (Ed.), LNCS, Vol. 674, pp. 71–90, Springer-Verlag, 1992.

[4] T. Murata, "Petri Nets: Properties, Analysis and Applications," *Proceedings IEEE*, Vol. 77, No. 4, pp. 541–580, 1989.

[5] C. Reutenauer, *Aspects Mathématiques des réseaux de Petri*, Masson, 1988. English edition: *The Mathematics of Petri Nets*, Prentice-Hall Intern., 1990.

[6] C. Seatzu, F. Balduzzi, A. Di Febbraro, A. Giua, "Decidability of single-rate hybrid Petri nets," *38th IEEE Conf. on Decision and Control*, (Phoenix, Arizona), Dec. 1999.

FORMULATION AND ANALYSIS OF AN ANALOG STATIC MODEL FOR URBAN PLANNING

R. De Lotto[1], A. Ferrara[2]
[1]DIET – Dipartimento di ingegneria edile e del territorio - University of Pavia
Via Ferrata 1 - 27100 Pavia - Italy
[2]DIS – Dipartimento di ingegneria informatica e sistemistica - University of Pavia
Via Ferrata 1 - 27100 Pavia - Italy

Abstract: The paper deals with the formulation and analysis of a static model aimed at being the kernel of possible computer aided design tools for urban planning. The key element of this model is a static representation of the transportation network connecting the considered facilities. Such a representation is based on an analogy of electric nature. As a result, conventional electric networks solvers can be used to determine the relevant features of the traffic flow and the impact of the facility location decision on the urban area.

Introduction

One of the major problems of urban planners is that of efficiently locating services or facilities in the urban area. In its optimal formulation, a location problem can be stated as follows: given m clients and n potential sites for locating prespecified facilities, taking into account the profit deriving from supplying the demand and the cost for setting up the facitities, select an optimal set of facility locations (see, for istance, [1], cap. 16).

Actually, the search for optimal solutions is the aim of operational research, while, as far as urban planning activities are concerned, the goal is often to develop suitable tools to be used as support systems in the decision process, during which a sub-optimal feasible solution is iteratively constructed. Relying on this basic consideration, the present paper formulates a static model oriented to the analysis and sub-optimal solution to location-like problems within a urban area. Making reference to a real urban context, the concept of "client" of a facility can be naturally replaced with the concept of "vehicle", since it is impossible to ignore the presence of an underlying transportation network through which the i-th client reaches the j-th facility [2]-[3].

The proposed model describes the road network as an electric network, enabling to evaluate the incremental traffic due to the presence of the facility. It describes each lane of a road as an oriented link, characterized by (time-varying) parameters, such as the travel time at given unsaturated conditions. Road intersections are represented as nodes with inflows and outflows. The propagation delays due to the presence of traffic lights ([4], cap. 1) and non homogeneous flows are also taken into account. In the paper, the analysis of the proposed model in terms of its sensitivity to possible variations of the network parameters is also discussed.

Facility location problems

The usual way to face facility location problems is to define a global cost function associated with a certain location decision to be minimized. In our case, assume to subdivide the considered urban region into N areas, to locate M facilities in certain precise positions, and to account for L possible types of transportation means. Then,

$$C = \sum_k \sum_j \sum_i p_i^k c_{ik} t_{ij}^k , \qquad 1 \le k \le L, 1 \le j \le M , 1 \le i \le N \tag{1}$$

(see [1]) where p_i^k is the user population in the i-th area using the transportation mean k, c_{ik} is the weighting factor associated with that type of transportation mean, t_{ij}^k is the time to reach, from the i-th area, the j-th facility with the transportation mean k.

Model formulation

The model proposed in this paper relies on the assumption that the dominant transportation mode ("dominant" in the sense that it primarily contributes to determine the access time), is the private vehicle.

Consider the problem relevant to M facilities S_j, $1 \leq j \leq M$, of a certain nature to be located in a urban context under the following assumptions: 1. each facility has the assigned capacity to serve C_j clients per hour; 2. the distribution of the potential clients over the urban territory is known (more precisely, the spatial distribution is discretized into elementary units called "cells"); 3. each cell i, $1 \leq i \leq N$, contains $p_i(t)$ potential clients, where such a quantity is a time function valued in number of clients per hour; 4. the i-th cell is centered in the i-th road intersection, $1 \leq i \leq N$, N being the total number of road intersections of the urban transportation network considered.

The p_i-th client's choice of the facility to reach is assumed to be dictated by the cost to access the facility. In the case of transportation by means of private vehicles, such a cost can be modelled as directly proportional to the vehicle travel time $t(i, j)$ from the i-th road intersection, from which the p_i-th client starts, to the j-th road intersection, where the facility is located. The global vehicle travel time is obviously the sum of the travel times associated with the links of the transportation network through which the client moves to reach the facility.

The urban transportation network can be represented by a graph with N nodes and a set of oriented links $l(i, j)$, $i \leq N$, $j \leq N$, connecting the i-th node with the j-th node, with the travel direction from i to j. Each link is marked by a label which defines the time-varying travel time law on the link itself as a function of the traffic volume $n_{ij}(t)$.

To further refine the model of the access to a facility it is necessary to consider that, relying on the model, the following aspects should be determinable: for any node, the time to access the nearest facility; the nodes "captured" by a facility, and the corresponding burden in terms of clients (this, in turn, allows one to identify the influence area of each facility delimited, on the nodes map, by the border lines connecting the nodes with associated longer access time); the number of clients reaching each facility; the induced traffic variation in each link of the transportation network; the total cost for the users which is implied by the selected facility location: this quantity can be computed by summing up all the access times associated with the nodes multiplied for the number of clients arriving from each node.

To determine the quantities indicated above, it is necessary to compute the travel time corresponding to each link of the considered transportation network, making reference to the particular traffic conditions in the time interval of interest. In [2], the travel time as a function of the traffic intensity, of the parameters which determine the traffic fluidity, and the possible presence of traffic lights is provided. Yet, in our case, the point is to evaluate the traffic variation due to the location of a certain facility in a certain area.

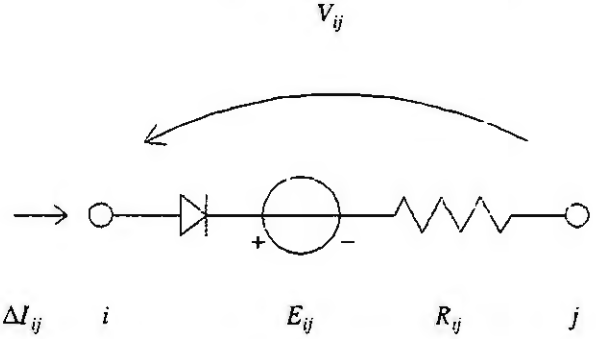


Figure 1: The electric equivalent of a link of the transportation network

Then, the mentioned function can be linearly approximated by the tangent in the operation point in question. More precisely, given the regular traffic in the considered link, one can determine the corresponding travel time $t_0(i, j)$. Then, the travel time after that the facility has been located can be written as

$$t(i, j) = t_0(i, j) + R[n(i, j) - n_0(i, j)] = t_0(i, j) + R_{ij}\Delta n(i, j) \tag{2}$$

where $\Delta n(i, j)$ is the traffic intensity induced by the considered facility and

$$R_{ij} = \left.\frac{dt(i,\ j)}{dn(i,\ j)}\right|_0 \tag{3}$$

Clearly, in searching the optimal facility location, it is the average access time to be crucial rather than the access time of a single client. This is the reason why a macroscopic continuous-time model turns out to be the correct choice. Moreover, since there are plenty of efficient commercial tools for the analysis of electrical networks, it seems natural to depict an electrical equivalent of each link of the transportation network, Fig. 1, in fact representing this latter as an electric network.

Note that, in the electric equivalent of a link, the presence of a diode guarantees that the current flows in a unique direction, and so does the traffic. The role of the voltage generator E_{ij} is to polarize the diode, thus representing the original travel time $t_0(i,\ j)$. The value of the resistor R_{ij} describes the dependence on the current I_{ij}, i.e., on the induced traffic intensity $\Delta n(i,\ j)$. Finally, the voltage V_{ij} represents the link travel time $t(i,\ j)$.

To determine the electric equivalent parameters corresponding to each link of the transportation network the following considerations can be made. The travel time along a urban road in regular traffic conditions is given by

$$t = \frac{3600\Lambda}{v} + t_s \tag{4}$$

where Λ is the length in Km of the patch between two subsequent road intersections, v is the mean speed (Km/h) of the vehicles, t_s is the additional time due to the presence of a traffic light. On the other hand, the mean speed v in a urban area can be obtained, experimentally, as a function of different parameters. For instance, for some Italian urban areas, one has

$$v = 31.1 + 2.8l - 1.2\rho - 12.8\bar{t}^2 - 10.4D - 1.4I - (53 \cdot 10^{-6} + 123 \cdot 10^{-6} X)\left(\frac{n}{\lambda}\right)^2 \tag{5}$$

where λ is the width of the carriageway for the considered direction having subtracted the width occupied by parked vehicles, l is the length of the carriageway, ρ is the slope expressed as a percent, $\bar{t}$ is the winding degree of the road in a normalized scale between 0 and 1, D is the degree of disturbance to circulation again valued in a normalized scale between 0 and 1, I is the number of intersections of the considered road, X is a quantity equal to 1 if the road does not give the possibility of surpassing, 0 otherwise, finally, n is the traffic flow in vehicle/h.

The time t_s due to the presence of a traffic light can be determined through the following approximate relationship

$$t_s = \frac{1}{2}T(1 - m)^2 + \frac{0.55}{ms} \cdot \frac{n}{ms - n} \tag{6}$$

where T is the duration of the traffic light cycle (sec), m is the ratio between the green time and the cycle duration T, s is the saturation flow (vehicles/sec) of the road, n is the vehicle flow (vehicles/sec). As a consequence, the considered piece of road will be characterized by

$$E = t_0 = t\big|_{n=n_0} \tag{7}$$

$$R = \left.\frac{dt}{dn}\right|_{n=n_0} \tag{8}$$

and, in the trasportation network, E_{ij}, R_{ij} are given, respectively, by E and R in (7)-(8) computed taking into account the peculiarities of the links i, j.

Modelling the clients population

Relying on the electrical modelling equivalent, the clients population can be modelled by means of ideal current generators which injects their current in the nodes of the transportation network where the barycenter of each cell is located. The values of the (clients) current are expressed in terms of the number of vehicles per hour (veh/h). They account for both the time distribution and the anagraphical and socioeconomic features of the clients population, this through a parameter related to the "appeal" of each facility. Note that with the term facility "appeal", we mean the capability that a certain type of facility has to attract the clients population. Data relevant to this capability can be acquired from national statistical studies centers (for instance, CENSIS). Generally, the available data provide the number of families attracted by the considered facility type, the number of persons for family unit, their distribution over the territory. From the modelling point of view, each facility, regarded as uncapacitated, is described by setting at a null potential the corresponding node where it is assumed to be located. The usage intensity of the considered facility is given by the sum of the currents entering such a node.

In alternative, as long as the facility has a limited capacity C_j, the node where the facility is placed is not directly put to mass, but its connection to the null potential level is that depicted in Fig. 2. In Fig. 2, V_j is equal to zero until $I_j \leq C_j$, that is up to the facility saturation. As soon as $I_j > C_j$, the diode is cut-off and V_j increases, practically re-distributing the clients among the other facilities, while keeping $I_j = C_j$. It is worth noting how this electrical effect resembles a waiting-in-queue time. More precisely, the queueing time of the j-th facility is modelled by the potential V_j.

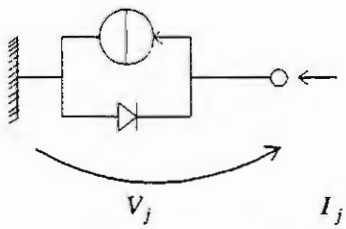


Figure 2: The electric equivalent of a capacitated facility located in the j-th node

Network solution

Given the parameters E_{ij} and R_{ij} for any link of the electric network representing the transportation network as a function of the operating point and of the dependence of the travel time on the traffic intensity, specified, in each node, the magnitude of the current generators modelling the clients population, and, finally, chosen a feasible location of the considered facilities, then, solving the network means to determine:

1. The voltage at each node on the basis of which it is possible to quantify the access times, the border lines and the consequent partition of the network nodes which identifies the influence areas of each facility. Note that the facility access time is given by the difference between the potential of the source node, i.e., the starting point, and the destination node, namely the node where the facility is located.
2. The current in each link of the network which represents the traffic variation induced by the facility.
3. The current in the node where the facility is located which describes the degree of utilization of the facility.

It is worth noting that the network solution, relying on the analogy between voltage at the nodes and travel times, and on the analogy between currents in the links and induced traffic, leads to the determination of the minimum of the cost function C indicated in (1), as recalled in [5]. Indeed, according to [6], in a circuit made up by resistors, independent voltage and current sources, the currents tend to reach a distribution such that the dissipated power is minimum.

The advantage of the proposed model relies on the possibility of using commercial electrical networks analyzers to determine the relevant parameters characterizing the behaviour of the clients population with respect to a certain choice of the facilities location. For instance, the problem of solving the electric networks which accounts for the effects on the underlying transportation network of the facility location choice belongs to the class of problems that can be easily dealt with by SPICE [7].

Examples of model sensitivity analysis

As an example, let us consider a urban road network taken from the map of Pavia, Italy. All the parameters are calculated relying on the data of the Traffic Urban Plan of the city [8]. The considered network has 36 nodes and 125 links. The location decision consists in placing a facility in Node 1 and another in Node 27, at opposite parts in the network. In order to evaluate the sensitivity of the overall network behaviour with respect to the variation of parameters of the links (e.g, the associated travel time), we have taken into account the mean access time to the facilities, this latter being evaluated as the weighted average value of the various access times.

Some results of the sensitivity evaluation of the mean access time to the facilities on the occurrence of a 20% variation of the travel time associated with the considered link are reported in Table 1. Note that, in the first row one has the indication of the considered link with source node and destination node, while the percentage variation of the studied global quantity is shown in the second row. Only the links with significant traffic flow towards the facilities have been considered in the analysis.

As a second example, the interruption of some links, due to accidents or work in progress on the road, has beem considered. For instance, the interruption of link 2-1 causes a percentage increment of the mean access time to the facilities of 26%, with a reduction of the access to the facility located in Node 1 of 22%, while, the the interruption of link 10-27 determines a percentage increment of the same global quantity of 14%, with a reduction of the access to the facility located in Node 27 of 2%

2-1	20-1	10-27	12-27	25-26	35-26	29-28	24-25	9-10	21-2
1.32%	1.06%	1.06%	0%	1.32%	0.8%	1.06%	1.06%	0.8%	0.89%

Table 1: Results of the sensitivity analysis

Conclusions

A static model oriented to the analysis of location problems from the point of view of urban planners has been presented in the paper. The proposed model provides a simple way to calculate the variation of traffic due to location decisions. For this reason, it could be part of a decision support system or a computer aided design tool for urban planning. The advantage of the modelling analogy relies on the possibility of using commercial electrical networks analyzers to determine the relevant parameters. A sensitivity analysis to evaluate the overall network behaviour on the occurence of parameter variations can be easily performed.

References

1. Dell'Amico, M., Maffioli, F. and Martello S. (Eds), Annotated bibliographies in Combinatorial Optimization. John Wiley and Sons, New York, 1997.

2. Cascetta, E., Metodi quantitativi per la pianificazione dei sistemi di trasporto (in Italian). CEDAM, Padova, 1990.

3. Gelmini, P., Modelli urbanistici di distribuzione (in Italian). CLUP, Milano, 1986.

4. Cantarella, G. E. and Festa, D. C. (Eds), Modelli e metodi per l'ingegneria del traffico (in Italian). F. Angeli, Milano, 1998.

5. Di Barba, P. and Savini, A., An optimization approach to the analysis and synthesis of circuits and fields. Computation in Electromagnetics, 420 (1996), 370-375.

6. Maxwell, J. C., A treatise on electricity and magnetism. Oxford Press, Oxford, 1892.

7. Nagel, L. W., SPICE: a computer program to simulate semiconductor circuits. ERL Memo M520, Electronics Research Laboratory, University of California, Berkeley, CA, 1975.

8. De Lotto, R., Un modello analogico per ottimizzare la dislocazione di servizi in un contesto urbano (in Italian). Internal Technical Report of DIET, University of Pavia, 1999.

The Flow of Large Crowds of Pedestrians

R. Hughes
Visiting: Delft Hydraulics
Delft, The Netherlands
Permanent Address: The University of Melbourne
Parkville, Victoria, Australia

Abstract. Despite popular belief the motion of a crowd is governed by well-defined rules of behaviour. These rules imply a set of coupled, non-linear, partial differential equations for the density and velocity potential for each type of pedestrian in the crowd. As may be expected, the solution of these equations may, in different regions of space, be supercritical or subcritical with the possibility of a shock wave separating the regions. Less predictable is the remarkable finding that these coupled, non-linear, time dependent equations are conformally mappable and this finding enables solutions to be obtained easily for both supercritical and subcritical flows.

Introduction.

Developing an understanding of the behaviour of a moving crowd of pedestrians is an important but largely neglected area of science and engineering. To an engineer there are two fundamentally distinct methods of modelling crowd motion. The first method (Lagrangian simulation) involves direct simulation. Each model pedestrian is given distinct properties and is followed throughout the domain. The second method (Eulerian simulation) is to grid the region of interest and study the flow without regard to specific model pedestrians. Individual model pedestrians are not followed, instead the number of model pedestrians in each grid box is studied. Both methods have their place and should be seen as complementing each other.

However, neither method can be considered to be scientifically acceptable as both methods require a preliminary specification of the path taken by pedestrians and this is given using the modellers intuition. The present study uses observed rules of pedestrian behaviour, which it formulates as hypotherses, to obtain a rational method for determining the path taken by pedestrians and the associated pedestrian density. Here, the determination of the path requires integration of partial differential equations. However, in chossing path an actual pedestrian generally does not do a calculation based on these or any other equations. An actual pedestrian draws upon experience of similar situations to choose a path. The results of this experience can be simulated by calcualtion.

Formulation

The present study seeks to rectify the problem of the choice of path by using a continuum model based on well defined observations of pedestrian behaviour, referred to here as hypotheses. For a single type of pedestrian the hypotheses are as follow:

Hypothesis 1 states that the speed, f, at which pedestrians walk is determined solely by the surrounding pedestrian flow and the behavioural characteristics of the pedestrians, that is, the velocity components (u, v) are given by

$$u = f(\rho)\hat{\phi}_x, \qquad v = f(\rho)\hat{\phi}_y \tag{1}$$

where $\hat{\phi}_x$ and $\hat{\phi}_y$ are direction cosines of the motion. This hypothesis is standard.

Hypothesis 2 states that pedestrians have a common sense of the task (called potential) they face to reach their common destination such that any two individuals at different locations having the same potential would see no advantage to either in changing places. There is no perceived advantage to a pedestrian of moving along a line of constant potential. Thus the motion of any pedestrian is in the direction perpendicular to the potential, that is, in the direction for which

$$\hat{\phi}_x = \frac{-\frac{\partial\phi}{\partial x}}{\sqrt{\left(\frac{\partial\phi}{\partial x}\right)^2 + \left(\frac{\partial\phi}{\partial y}\right)^2}} \qquad \hat{\phi}_y = \frac{-\frac{\partial\phi}{\partial y}}{\sqrt{\left(\frac{\partial\phi}{\partial x}\right)^2 + \left(\frac{\partial\phi}{\partial y}\right)^2}}, \tag{2}$$

where ϕ is the potential. This hypothesis is not appropriate to vehicular traffic but appears to be applicable to pedestrian flows where pedestrians can visually assess the situation.

Hypothesis 3 states that pedestrians seek to minimize their (accurately) estimated travel time, but temper this behaviour to avoid extremely high densities.

As two pedestrians on a given potential must both be at the same new potential as each other at some later time (noting time is a measure of potential by Hypothesis 3), the distance between potentials must be proportional to pedestrian speed irrespective of the path followed by a pedestrian. Thus we write

$$\frac{1}{\sqrt{\left(\frac{\partial\phi}{\partial x}\right)^2+\left(\frac{\partial\phi}{\partial y}\right)^2}}=\sqrt{u^2+v^2} \tag{3}$$

where ϕ has been scaled appropriately.

Equations (1), (2),(3) and the usual equation of continuity combine to form the governing equations for pedestrian flow

$$-\frac{\partial\rho}{\partial t}+\frac{\partial}{\partial x}\left(\rho f^2(\rho)\frac{\partial\phi}{\partial x}\right)+\frac{\partial}{\partial y}\left(\rho f^2(\rho)\frac{\partial\phi}{\partial y}\right)=0 \tag{4}$$

and

$$g(\rho)f(\rho)=\frac{1}{\sqrt{\left(\frac{\partial\phi}{\partial x}\right)^2+\left(\frac{\partial\phi}{\partial y}\right)^2}}, \tag{5}$$

where the factor $g(\rho)$ accounts for the tempering behaviour at high densities as referred to in Hypothesis 3. This formulation can be easily extended to crowds involving multiple pedestrian types. Observations of pedestrian motion suggest that $f(\rho)$ and $g(\rho)$ can be approximated by

$$f(\rho) = \begin{cases} A, & \rho \le \rho_{\text{trans}} \\ A\sqrt{\dfrac{\rho_{\text{trans}}}{\rho}}, & \rho_{\text{trans}} < \rho \le \rho_{\text{crit}} \\ A\sqrt{\dfrac{\rho_{\text{trans}}\rho_{\text{crit}}(\rho_{\max}-\rho)}{\rho^2(\rho_{\max}-\rho_{\text{crit}})}}, & \rho_{\text{crit}} < \rho \le \rho_{\max} \end{cases} \tag{6}$$

and

$$g(\rho) = \begin{cases} 1, & \rho \le \rho_{\text{crit}} \\ \dfrac{\rho}{\rho_{\max}-\rho}, & \rho_{\text{crit}} < \rho \le \rho_{\max} \end{cases} \tag{7}$$

where $A = 1.4$ ms^{-1}, $\rho_{\text{trans}} = 0.8$ ms^{-2}, $\rho_{\text{crit}} = 3.0$ ms^{-2}, and $\rho_{\max} = 5.0$ ms^{-2} typically. The detailed forms of (6) and (7) depend on the properties of the pedestrians being studied. However, the forms given are close to the behaviour observed for most crowds and are mathematically convenient. At low densities, less than ρ_{trans}, the speed of pedestrians is constant. For these densities, the speed is limited by the ability of pedestrians to move there limbs quickly. For densities greater than ρ_{trans}, the speed of pedestrians is limited by fear of collision, and interference between pedestrians occurs. The speed of pedestrians decreases with in increased density until pedlock occurs at a density of $\rho_{\max}$. Flows of pedestrians at conditions near pedlock are often frightening to those involved and hence evasive action is taken by many pedestrians, thereby increasing the function $g(\rho)$ above unity.

Consideration of the form of (6) shows that while $f(\rho)$ is continuous, $f'(\rho)$ is not continuous at $\rho = \rho_{\text{trans}}$, or $\rho = \rho_{\text{crit}}$. Avoiding issues associated with the behaviour at $\rho = \rho_{\text{crit}}$, it is clear that $(\rho f(\rho))'$ is positive for $\rho < \rho_{\text{crit}}$ and negative for $\rho > \rho_{\text{crit}}$. Thus following [1], disturbances are swept downstream for $\rho < \rho_{\text{crit}}$ but propogate upstream for $\rho > \rho_{\text{crit}}$. Critical conditions therefore occur when $\rho = \rho_{\text{crit}}$.

A Solution Method

An interesting method of solving (4) and (5) is conformal mapping. Despite the time dependent nature of the formulation of (4) and its non-linearity, (4) and (5) are conformally mappable. To understand the application of conformal mapping it is necessary to consider the behaviour for $\rho \le \rho_{\text{crit}}$, $\rho_{\text{trans}} < \rho \le \rho_{\text{crit}}$ and $\rho_{\text{crit}} < \rho \le \rho_{\max}$ separately.

For $\rho \leq \rho_{crit}$, pedestrians walk with a constant speed irrespective of density. They walk in straight lines between their origin and destination, only changing direction when forced to do so by boundary geometry. The path taken is the shortest path.

For $\rho_{trans} < \rho \leq \rho_{crit}$, (4) and (5) are conformally mappable as formulated in terms of ϕ and ρ. Under conformal mapping ϕ remains unchanged but ρ scales as the square of the Jacobian of the map. By (5), the speed of pedestrians, f, also scales but as the inverse of the Jacobian.

For $\rho_{crit} < \rho \leq \rho_{max}$, (4) and (5) are again conformally mappable. Under conformal mapping ϕ remains unchanged but $(\rho_{max} - \rho)$ scales as the Jacobian of the map. For these densities the motion of 'holes' in the crowd are of more physical significance than the motion of pedestrians in the crowd and so the scaling of $(\rho_{max} - \rho)$ rather than ρ is not surprising.

The application of this solution technique is straightforward, and will be illustrated in the talk accompanying this manuscript. The only difficulty involves the choice of boundary conditions. For example in studying the motion of pilgrims over the Jamarat bridge near Mecca the attitude of pilgrims to their objective is critical. Pilgrims are required to stone three pillars in turn. If pilgrims are assumed to set as their objective the stoning of the next pillar then a crowd of pilgrims is predicted to develop in front of each pillar, with low density on the sides of each pillar. However, if pilgrims were to see the leaving of the whole site as their objective, there would be a low density in front of each pillar and high density at the sides as pilgrims, positioning themselves to move to the next pillar, throw their stones as they walk past en route to the next pillar. In practice the former behaviour is observed.

Extensions

There are two important extensions to this work that need to be noted. Firstly, it has been assumed that the speed of pedestrians is only a function of the density of pedestrians. Many situations exist where the speed, f, also depends on position because of non-uniformity of the surface on which the pedestrians walk. In such cases (4) and (5) are still correct, but they are no longer conformally mappable. Secondly, in many cases more than one type of pedestrian are involved. In such cases (4) and (5) hold for each type of pedestrian with two equations for each pedestrian type. Surprisingly, it is well established from observations that the only change in the speed $f(\rho)$ is that the ρ now refers to the total density of pedestrians, not the density of each type of pedestrian. This unexpected behaviour results from the way pedestrian crowds walk through each other.

Conclusions

Despite popular belief the behaviour of pedestrians is rational and easily formulated mathematically. The resulting equations have been shown to be highly non-linear. However, despite this nonlinearity and possible time dependence, the equations have the remarkable property of being conformally mappable. The greatest difficulty in applying this formulation involves the appropriate choice of boundary conditions to match the psychological state of the pedestrians. As shown earlier, the psychological state of pedestrians can completely change the flow pattern, as illustrated by a case study of flow over the Jamarat bridge. There is great scope for improving the safety of pedestrians at many major events. As the location of pedestrian accidents can often be accurately predicted, those changes required for safety can be implemented inexpensively. Fascinating problems involving the prevention of major accidents await anyone who chooses to persue a study of these crowds.

References

I. Lighthill, M.J. and Whitham, G.B., On Kinematic Waves: I Flood Movement in Long Rivers; II Theory of Traffic Flow on Long Crowded Roads. In: Proc. Roy. Soc. A., 229, 1955, 281 – 345.

MODELLING OF PRODUCT RECYCLING CHAINS

U. Kleineidam, A.J.D. Lambert, J. Banens, J. Kok, R.J.J. van Heijningen
Eindhoven University of Technology
P.O. Box 513, 5600 MB Eindhoven, The Netherlands
email: u.kleineidam@tm.tue.nl

Abstract. The cycles in substance chains have become a major aspect of environmental policy because recycling of discarded products is an effective method of diminishing the discharge of wastes and decreasing the depletion of resources. In this paper, a modelling method is described for product chains including recycling. It consists of elementary models of standard production operations, connected by market models. Companies and markets in the chain are described by a state space model as used in control theory. The model is illustrated by the example of the Dutch paper chain. A regulation which has recently been applied to this chain — the supply and purchase agreements agreed upon in the paper fibre convenant — is simulated.

Introduction

In an attempt to answer the call for sustainable development, recycling of products is gaining importance. The introduction of cycles in product chains reduces the amount of waste and the use of resources. On the other hand, it implies that product chains are becoming increasingly complicated. It is of great interest to both firms and public authorities to obtain insight into the functioning of such chains.

This paper presents a modelling method for product chains. The entire chain associated with a product is considered according to a 'cradle to grave' approach, i.e. from the extraction of raw materials to the disposal of waste, including the reuse of parts and the recycling of materials. It consists of standard production operations, connected by market modules, and allows us to investigate essential properties of the chains concerning their dynamical behaviour.

State space models of macro-economic systems, such as national economies, have been used earlier in order to determine an optimal policy, e.g. in [4]. Unlike these macro-economic systems, product-process chains, as analysed here, are meso-economic systems which typically describe a sector or branch of industry, e.g. the paper or the steel industry. Another field where control theory is used is the field of logistics [2]. Controllers are designed in order to minimise the costs associated with inventory and production. Optimal production policies are determined under conditions such as deteriorating products [1] and bounded production [3].

The company model

In the product chains considered here, an entire industrial sector is modelled. Various interacting actors exist just at each life phase of the chain (various producers, various consumers, etc.). For the sake of model simplicity, it is assumed here that the actors in each phase can be modelled by a single aggregated 'company', a usual technique in economic modelling [7]. The product chain can then be considered as a network of such basic units linked to each other by markets. Figure 1 shows a small flowchart of such an aggregated company. Since only the main flows associated with a product are considered, only markets corresponding to the main resources and the main product are included in the model.

The company's basic activity consists of the transformation of resources into products. It is represented by a square. Resources and products are stored in inventories, represented by triangles. Note that, in the limiting case, one or both inventory levels in Figure 1 may constantly be zero. The decision to hold inventories or not depends on the kind of product the company produces. It is a consequence of its production decision system (production to stock, production to order, etc.) [10]. If both inventories are zero, for example, an ideal 'Just In Time' system is represented.

The company model which is derived here, includes the company's decision on the quantity of resources it wants to purchase q_d, and on the quantity of products it is willing to sell q_s, see Figure 2a. This decision is taken on the basis of the available information, i.e., the price of resources p_1, the price of products p_2, the actual amount of purchased resources e_1, and the amount of sold products e_2.

Abiding by the law of mass conservation the in- and outflows of mass to the inventories in Figure 1 need to balance their accumulation. The mass-balance applied to the second inventory states that the accumulation of products in stock ($\dot{g}_2$ in kg/year) equals the the production q_p (in kg/year) minus

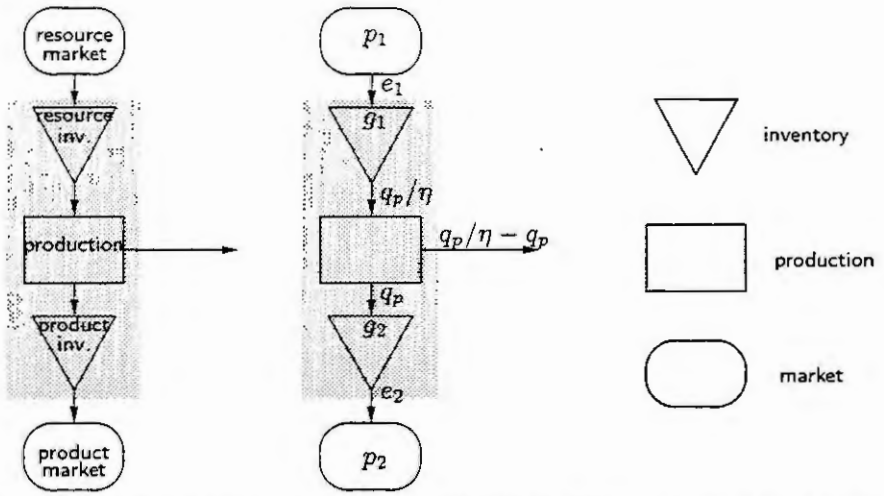


Figure 1: The product flows in a company with two inventories. The left figure shows the transformation processes and the product flows, and the right figure shows the corresponding variables.

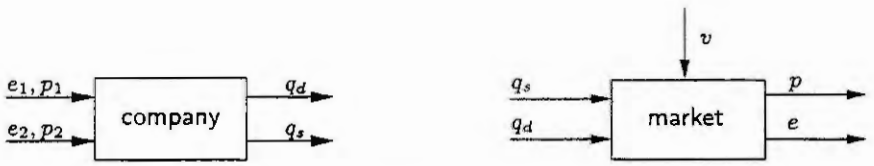


Figure 2: The inputs and outputs of the company subsystem(a) and of the Walrasian market (b)

the outflow, which equals the flow sold on market 2 (e_2 in kg/year). Accordingly, the accumulation in inventory 1 equals the flow of purchased resources e_1 minus the amount of resources needed for production. We obtain the balance equation for inventory 1 and 2

$$\eta \dot{g}_1 = \eta e_1 - q_p \qquad \dot{g}_2 = q_p - e_2 \tag{1}$$

Here the production efficiency η (in kg product/kg resource) is taken into account. For one unit of final product, η units of resources are needed, consequently $\eta \leq 1$. The main activity of a producing company is the transformation of resources into products. The production decision can be translated into the produced quantity q_p of products being a function of the product price p_1, the resource price p_2 (both in €/kg), the resource stock g_1 and the product stock g_2. Here a linear relationship is assumed.

The company's decisions on demand q_d and supply q_s depend on the production as well as on the resource and product stocks. The company is willing to purchase the quantity of resources q_d (in kg/year) on the resource market. In general, the supply of products q_s (in kg/year) on the product market is a function of the product stock g_2 and the price p_2 that the company can obtain on this market. The company model can be written in the usual state space representation of control theory

$$\begin{aligned} \dot{x} &= f(x,u) \\ y &= g(x,u) \end{aligned} \quad \text{with} \quad x = [\eta g_1 \; g_2]^T, \quad u = [p_1 \; p_2 \; e_1 \; e_2]^T, \quad y = [q_d \; q_s]^T \tag{2}$$

The market model

Markets are considered here as virtual places, in which a trade-off between demand and supply takes place, and which are governed by a price mechanism. In the standard economic theory of a competitive market, a supply function, which increases with an increasing price, and a demand curve, which decreases with an increasing price, are considered. The equilibrium price is given by the intersection of the two curves. However, markets are generally not in equilibrium.

Market disequilibrium is much less understood. In the theory of exchange, a Walrasian mechanism is considered [8]. The price adjustment is based on the simple formula originally proposed by Walras,

stating that the time derivative of the market price p equals some adjustment parameter ρ (in €/kg) times the excess demand. The signals corresponding to the market are depicted in Figure 2b. The totals of market demand q_d and supply q_s are attained by summing the supplies and demands of the individual actors. The state equation of the market is

$$\dot{p} = \rho(q_d - q_s) \tag{3}$$

The real product flow, or trade e, is given by the minimum of demand and supply, because a supplier cannot sell more than he supplies and a demander will not purchase more than he needs

$$e = \min(q_s, q_d) \tag{4}$$

Trade from and to the actors outside the considered system constitute disturbances of the market and are denoted by v. A positive v denotes imports to the system, and exports are represented by a negative v. If we take these external influences into account, the total trade is given by $e = \min(q_s + v, q_d) - \min(0, v)$.

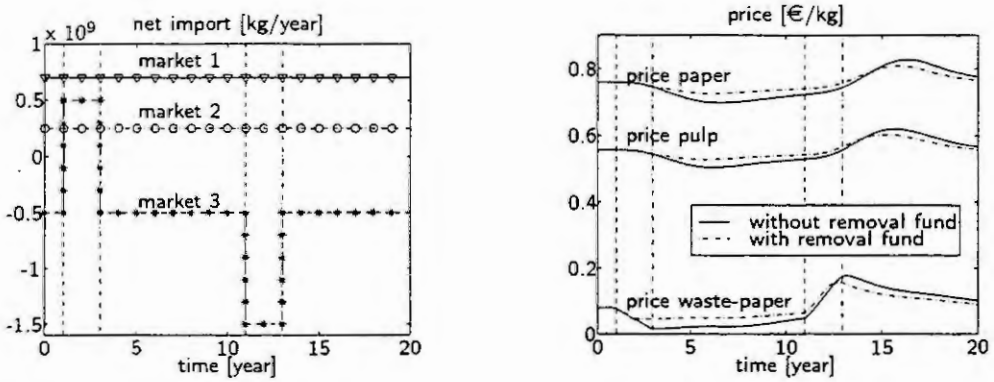
The chain model

A model of an entire product chain is built up by combining the company and market models. The outputs of the market models — the prices and trades — constitute inputs to the company models and vice versa. As an example of a closed chain, we consider the paper chain. Paper is a substance made from fibrous cellulose material. Paper and paper products are used in various products and groups of products in the whole economy. The cellulose pulp used in the paper manufacturing process is either a primary raw material such as pulp from wood or a secondary raw material such as pulp from waste-paper.

The paper industry either buys primary pulp or pulp from waste-paper (or a combination of both) on the pulp market. Subsequently, the paper and cardboard industry supplies the crude paper it has produced to the paper and cardboard market. The paper and cardboard market also experiences supply from outside the chain by foreign suppliers. Some of the bulk paper is bought by the paper and cardboard product industry, converting the crude paper and cardboard into products of paper and cardboard (printed paper, cardboard boxes for packaging, etc.). Currently, wastes from offices, shops, and companies are transferred to the waste-paper market. Household wastes are collected by the municipality services and a large and entangled network of associations. Both flows form a supply to the waste-paper market where the waste-paper industry buys its resources. The sorted and processed waste-paper is supplied to the pulp market [9].

The paper chain consists of three markets (for pulp, paper, and waste-paper) and three chain actors (a paper producer, a paper consumer, and a waste-paper recycler). The state vector of this chain is thus nine-dimensional. Due to the nonlinearities in the market models, the chain system has a piecewise linear structure. The parameters in the model, such as the price adjustment constants ρ and the efficiencies η, are estimated by minimising the least squares difference between the model outputs and the given data. For the problem of parameter estimation, the development of the prices, of the imports to, and of the exports from the chain can be used. They have been observed for the Dutch paper chain by the Dutch Statistical Office for a long time [5].

As an example to illustrate the chain model, we present some results of the simulation of a regulation scheme which has recently been installed in the Dutch paper chain; the so-called paper fibre convenant. This convenant was introduced as a voluntary agreement between the paper chain actors and the government. It demands the entire chain to work on prevention and recycling of the waste-paper stream. It must be noted that the agreements are not only supposed to increase the amount of recycled paper but that they are also in the economic interest of the chain. In the past, waste-paper supply and demand fluctuated heavily [6], which had consequences for the whole chain. Due to low waste-paper prices, for example, the waste-paper collectors were no longer willing to collect waste-paper. This led to a blocking of the chain. It is expected that the agreements of the convenant will be able to moderate these fluctuations. Moreover, a less fluctuating waste-paper supply means that recycling capacity can be better used, which is economically advantageous. The agreements on purchase guarantees, included in the convenant, are put into practice by installing a so-called *removal fund*. If the waste-paper price drops below the cost price, this fund will buy the excess waste-paper and will pay the cost price. Here, the system after implementing the convenant is compared with the system without convenant. In this example, the equilibrium situation is taken as initial situation. The net imports to the chain used in the



(a) (b)

Figure 3: The imports (a) and the prices (b) of the paper chain with and without paper convenant

simulation of the paper chain model are depicted in Figure 3a. The reactions of a chain with removal fund and of a chain without removal fund are depicted in Figure 3b. This figure shows that, in both situations, all the prices drop after one year due to the excess waste-paper supply. The increase of export after 11 years, on the other hand, causes the prices to rise. It can be seen, for this example, that in the case of the removal fund, the prices fluctuate less than without removal fund.

Conclusions

It has been demonstrated that a modelling method based on control theory provides insight into the dynamics of product chains. A modular approach has been chosen: Chains are composed of modules of both production processes and markets. The method is applied to the Dutch paper chain, particularly for investigating the influence of convenants on the chain. It is demonstrated that this method is able to reproduce some aspects of past chain behaviour and might be a useful tool for decision support on future policy.

References

[1] A. Andijani and M. Al-Dajani. Analysis of deteriorating inventory/production systems using a linear quadratic regulator. *European Journal of Operational Research*, 106:82–89, 1998.

[2] S. Axsäter. Control theory concepts in production and inventory control. *International Journal of Systems Science*, 16(2):161–169, 1985.

[3] A. Bradshaw and Y. Erol. Control policies for production inventory systems. *International Journal of Systems Science*, 11:947–959, 1980.

[4] M.B. Chiarolla and U.G. Haussmann. Optimal control of inflation: a central bank problem. *SIAM Journal of Control and Optimization*, 3:1099–1132, 1998.

[5] Dutch Statistical Office (CBS). *Jaarstatistiek van de Buitenlandse Handel (Yearly Statistics of Foreign Trade)*. CBS, Voorburg, NL, yearly. in Dutch.

[6] A. Huttunen and T. Pirttila. Price dynamics on the recovered paper market. *International Journal of Production Economics*, 56–57:261–273, 1998.

[7] Y.-K. Ng. *Meso-Economics: A Micro-Macro Analysis*. Harvester Wheatsheaf, Sydney, 1986.

[8] A. Takayama. *Mathematical Economics*. Cambridge University Press, 1985.

[9] Y. Virtanen and S. Nilsson. *Environmental Impacts of Waste Paper Recycling*. Earthscan Publications, London, 1993.

[10] W.L. Winston. *Operations Research: Applications and Algorithms*. PWS-Kent, Boston, 1991.

MODEL OF DISTURBED PRODUCTION SYSTEM

S.Hadji and J. Favrel
Laboratoire PRISMa, INSA de Lyon
20, Avenue Albert Einstein, 69621 Villeurbanne cedex, France
Tel: (33)-4-72-43-84-87 fax: (33)-4-72-43-85-18 e-mail : shadji@ifhpserv.insa-lyon.fr

Abstract. We propose in this article to attach to the nominal supervisory control reactive modules to disturbances in a goal of functioning maintenance under disturbance. This type of reactivity remains adapted to the engaging of the degraded functioning from the detection of a disturbance. The insurance of production flow continuity is based on the commutation of adapted modes to the detected dysfunction. It means, the replacement of a module by an other in case of anomaly and, insurance inter-mode passages (nominal functioning, degraded functioning). If an event qualified as relevant appears, the behaviour of the system goes from nominal functioning to a type of specified degraded functioning. Our first contribution tries to formalise change mechanisms of functioning mode and, to validate formally the nominal functioning and others. Our second contribution will consist in developing an approach based on the principle of supervisors commutation, for the robust control of discreet-events systems (DES).

I- General concepts

The idea of the structure is to couple, not only one, but several supervisors to the process.
Indeed, our motivation is linked to the unpredictable and exceptional character of perturbations and, to the necessity to process critical perturbations not taken in account a priori. Generally, most current perturbations are integrated to the nominal functioning. Expected in advance, these perturbations lose in a certain manner their abnormal character. One of the main difficulties met by the utilisation of such approach called integrated approach, resides in the fact that the designer must a priori envisage all symptoms being able to appear and anticipate their processing. Including exhaustively all disturbances from the conception of the control behaved to a possibly useless overload. Obviously, it is necessary to take account some disturbances.

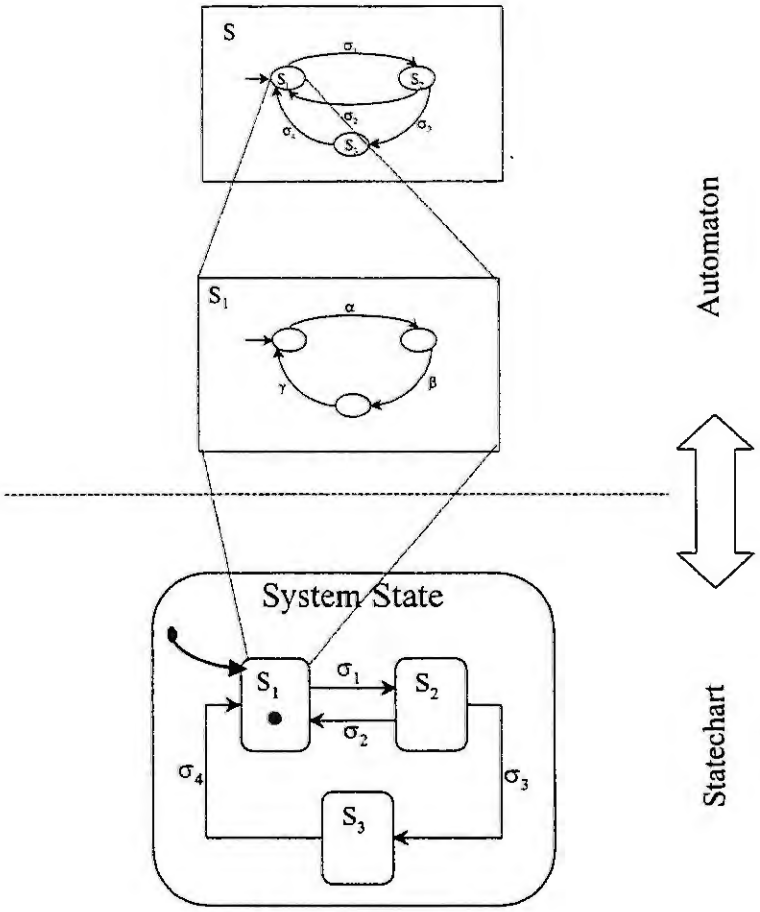
Supervisors do not work simultaneously, contrarily to the modular supervision. They work towards to the global supervision of the process.

A such approach is interesting in the sense where it simplifies complex problem resolution. Indeed, as the complexity of the different supervisors is lesser than the unique supervisor complexity of a centralised supervision. They can more easily synthesise, on the one hand, and on the other hand, in the case where a specification has to be added or modified, it is not necessary to modify the global strategy of control. Only the calculated supervisor for this specification will be added or modified. Each supervisor acts alone on the process in a certain context. It means that in this context, it has a global and total information. However, the global supervision results from the work of the whole supervisors, independent some of others.

The structure that we propose is composed of nominal supervisors and degraded supervisors. The different blocks are overlapped. Each sub-model is a supervisor that is going to control the system in a certain context. The replacement of a block by an other is made by commutation following the occurrence of an event characterised as relevant.

The automaton is a graphical language used to specify the control of discreet-event systems [1]. When trying to use this tool beyond its preferential application field for example, to specify working modes, we are rapidly confronted with problems posed by the complexity of what it is necessary to describe and by the important size of descriptions. It becomes essential to structure the specification bringing us to exploit the concept of hierarchy.

The structure that we propose is given in the following figure.



Let σi (i = 1..n; n∈Z), events that switches the system from functioning to another.

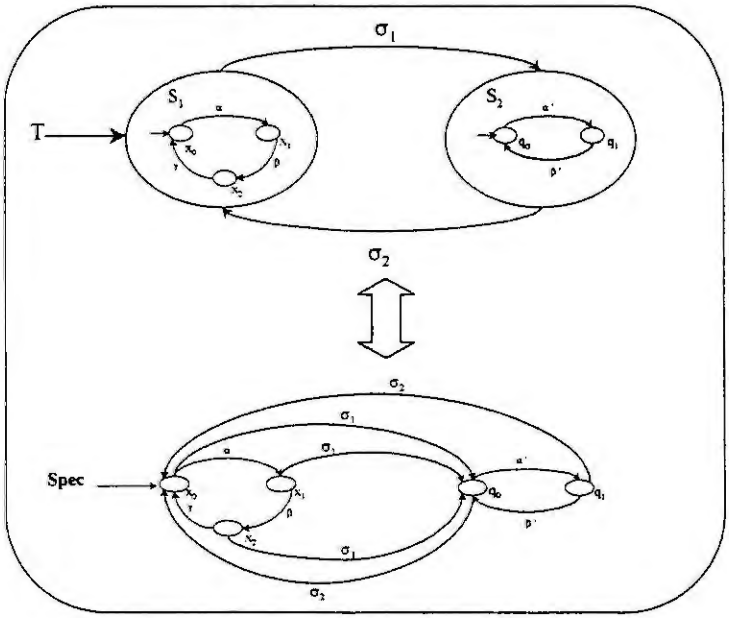
In the structure that we propose, our first constraint is that we have to be able to commute from a supervisor to an other. It means to know how to switch from a set of states or activities to an other set of states or activities, during the commutation of the different blocks. We should be able to empty an automaton (to block it) and to activate an other automaton, that will control to one's turn the system. This is not allowed by automaton. We propose to associate to automatons, a related graphical language, the Statechart [5] [6], that offers an interesting representation of the notion of hierarchy. According to our objectives, Statechart is a more effective tool. The Statechart allows us to deactivate some modules, to launch some others according to the workshop configuration.

Thus we define a precise job framework for each of the two languages.
- Statechart defines the working modes and describes the logic of evolution between these modes.
- Automatons refine the waited behavior when a mode is reached and realise mainly the effective control. In other words the Statechart insures the passage from the nominal mode to the degraded mode (and inversely). When the nominal mode or the degraded mode is reached, automatons insure the supervised control, following the classic theory of Ramadge-Wonham (RW) [2] [3] [4].

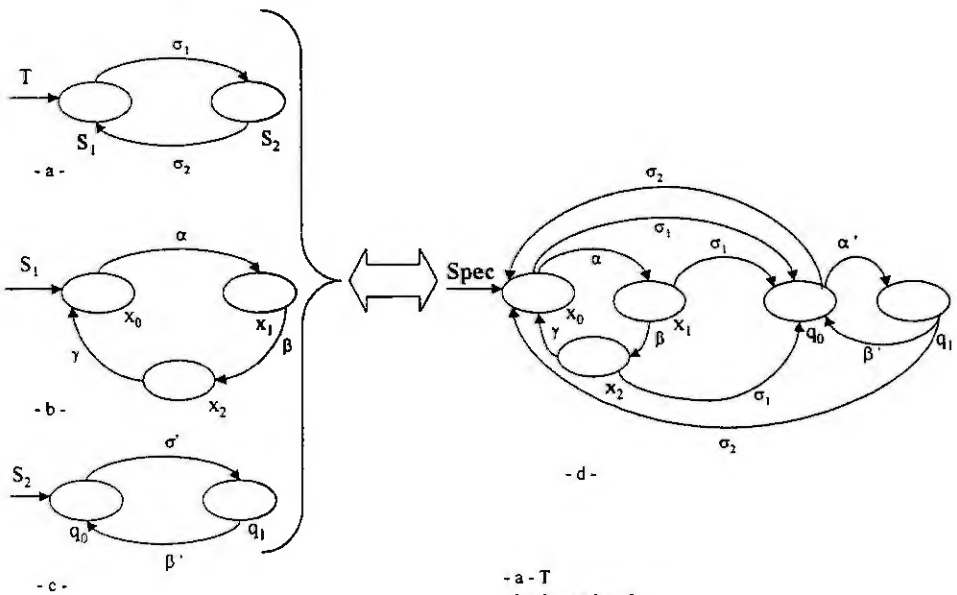
II - Alphabet partition

Our concept is based mainly on the partition of the global alphabet. Let us have Σn, the set of events of the nominal working and Σ_T the set of events of the degraded working, with $\Sigma_n \cap \Sigma_T = \varnothing$ ($\Sigma d = \{\sigma_i\}$) It is true that this classification comes naturally, by simple knowledge of the physical process. We have to validate two different

levels. The first level will be interested only in relevant events corresponding to a detection of degradation of the nominal working. Thus only breakdown and repair events will be represented. The second level is a refinement of the first level at each state. The validation of the second level corresponds to the each refined state validation of the first level. As alphabets and dynamics are non-contiguous, the sum of separated proofs is equivalent to the global proof. The first level of the model, qualified "high level", is therefore a Statechart (t). The "low level" is an active state of the Statechart (S_1 and S_2), and therefore modelled by an automaton.



Our objective is to validate the global structure. This partition implies that the global proof (Spec) is equal to the sum of separated proofs (S_1, S_2 and the Statechart T).



- a - T
- b - Supervisor S_1
- c - Supervisor S_2
- d - Automaton Spec (S_1o S_2 mod [T])

III Conclusion

We have proposed to use jointly two graphical languages [7], Statechart and Automatons, to specify the supervisory robust control of automated production systems. We have defined a precise action field for each of them :

- The Statechart serves to describe working modes and the logic of mode changes;

- Automatons have served to describe the detailed behaviour of the control when it will have reached a particular mode following the occurrence of particular event. They will be used, in a first time, to refine "leaf" states (elementary states) of the Statechart defining modes and, in a second time to realise the effective control.

This approach presents many advantages, it allows in particular:

- to set up working modes of the studied system by showing graphically structural bonds (modes / sub-modes) ;

- to exploit the force of the Statechart thanks to its primitive that are very simple to use.

- to describe the logic of mode change in a very synthetic manner and very expressive manner, that thanks to the power of evolution mechanisms associated to the concept of hierarchy and to the generalisation of the notion of transition. For instance, it becomes possible to describe in a very concise manner that the progress of a mode will be interrupted if an event acting at a superior level occurs;

- to model in a formal manner, the mode changes on automatons that manage the progress of these modes.

References :

1. Wonham, W.M., "Notes on control of Discrete-Event", ECE 1636F/1637S, System Control Group, University of Toronto, 1994-1995.

2. Ramadge, J.G. and Wonham, W.M., "Modular supervisory control of discrete event systems", Mathematics control, Shinals and Systems, 1988, Vol.1, n°1, pp. 13-30.

3. Ramadge, J.G. and Wonham, W.M., "The control of discrete event systems", IEEE Transaction on Automatic control, 1989, Vol. n°1, pp. 81-98.

4. Ramadge, J.G. and Wonham, W.M., "Supervisory control of class of discrete event processes", SIAM J. Control and Optimisation, 1987, Vol. n°25, pp. 206-230.

5. Sahraoui, A.E.K. and Ould Kadour, N., "on SART and Statecharts for reactive systems specification", 12th IFAC Congress, Sydney, 1993, vol5, pp317-320.

6. Harel, D., "Statecharts, a visual formalism for complex systems", Science of Computer Programming, North-Holland, n°8,1987, pp. 231-274.

7. Hadji, S. and J. Favrel, Statechart and Automatons in the supervisory Robust Control. *International Conference on Advances in Production Management Systems*, pp171-177.

SIMULATION MODEL FOR OPTIMIZATION OF RESOURCES ALLOCATION IN QUEUING NETWORKS

O. Zaikin [1], A. Dolgui [2], P. Kraszewski [1]
[1] Technical University of Szczecin,
ul. Zolnierska 49, 71-210, Szczecin, Poland
[2] University of Technology of Troyes,
12, rue Marie Curie, 10010, Troyes Cedex, France

Abstract. In this paper, we examine an integrated analytical and simulation approach for resources assignment in network models. We study the models with star configuration. The task is formulated as choice of multi-channel queuing system parameters. The analytical calculation and simulation model is proposed for analysis and searching of optimal solution.

Introduction

Several real systems as distributed computer, telecommunication, logistics and production systems can be described by queuing network (QN) models. For example, the applications of queuing models for productions systems are given in [3], for logistics –in [9], for computer networks and telecommunication –in [4].

We develop an integrated simulation and analytical calculation approach for optimization of resources allocation in the QN, which consists of following steps:

i. choice of basic queuing network models and their analytical verification,
ii. incremental construction of complete model while using the basic models,
iii. use of simulation techniques for validation of the complete model,
iv. jointly use of simulation and iterative algorithms for parameters optimization of the model, obtained in previous steps.

Several concepts of proposed approach have been developed in our previous works. Different methods of queuing model analysis have investigated in [10]. Several simulation techniques and languages have been used. An effectiveness method of stochastic optimization has been developed [5, 6]. Various examples of industrial systems design have been treated, for example [11, 12].

Problem statement

QN may be represented as oriented graph. Each vertex of the graph is a processing node, which is a set of one type processing equipment, performing some kind of operations. Each arc of graph represents flow process between two corresponding processing nodes. Hence, each processing node includes a number of servers, operating in parallel, input buffer and output buffer. Such structure permits to realize the parallel servicing process of several jobs simultaneously.

Distribution laws of jobs arriving and service time characterize the flow process. Now, let us define the rates of arrival and servicing of the flow process, that mean the number of jobs, incoming or served per time unit.

Let's examine the particular case of QN, represented by the oriented graph with star configuration. It means, that there is a set of processing nodes, each of them consists on some number of private servers, operating in parallel the arriving jobs with given productivity. Moreover, there is a network-center node with some number of common ('leased') servers, which can serve the jobs of all the flow processes with the same productivity for all leased servers.

The jobs, arriving at a processing node, are served in 'first come-first served' discipline. If all the private servers of the processing node are occupied, the incoming job goes to network-center node. If all the leased servers of network-center node are occupied, the incoming job is located in waiting queue of its processing node and in a given time interval (so called repeating time) returns for servicing. There are no restrictions on the number of jobs at waiting queue. It would be taken into account, those rates of jobs arrival and servicing, costs of the idle time of the servers at each processing node and network-center node may differ considerably.

Therefore the formulation of the resources allocation task in QN is the following problem.

For given set of processing nodes and flow processes and their parameters, it *is necessary to assign* a number of private servers for each processing node and leased (common) servers for network-center node, *providing* two alternative criteria:

1. The total servicing time of all the arriving jobs at all the processing nodes during period of optimization

$$T_{\Sigma} = \sum_{f} \lambda_f \, \tilde{\tau}_f \, T_0 \to \min, \tag{1}$$

where $f \in F$ is the index of flow process, λ_f is the rate of arrival jobs at the flow process 'f', $\tilde{\tau}_f$ is the average servicing time of a job at processing node for flow f, T_0 is the period of optimization.

2. Total costs of the idle time of all servers (private and leased) during period of optimization

$$C_{\Sigma} = \left[(1 - \frac{\tilde{\lambda}_C}{\mu_C N_C}) \gamma_C + \sum_{s} (1 - \frac{\tilde{\lambda}_s}{\mu_s N_s}) \gamma_s \right] T_0 \to \min, \tag{2}$$

where $s \in S$ is the index of processing node, $\tilde{\lambda}_s$ is the rate of jobs, incoming at the processing node s, μ_s is the rate of servicing of a private server at node s, N_s is the a number of private servers, assigned for node s, γ_s is the cost of the idle time unit for server s, $\tilde{\lambda}_C$ is the rate of jobs, incoming at the network center for servicing, N_C is the a number of leased servers, assigned for the network-center node, γ_C is the cost of the idle time unit for leased server.

The represented above criteria depend on control parameters in opposite way. Therefore, it is necessary to define such values of those ones, which provide some reasonable levels of both criteria. There are known some modes of searching of compromise decision, such as: a) choice of one from both criteria as the restriction, b) introducing of weight coefficients, c) Pareto optimization.

The solutions methods

At general case, the formulated task has not an analytical solution. However, primary evaluation can be obtained on the base of queuing systems theory [9]. On the Fig.1 the structure of multi-server queuing system with different kinds of servers and unlimited capacities of input (IB) and output (OB) buffers is represented.

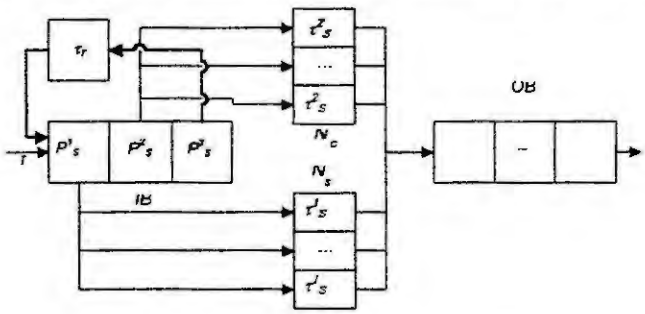


Fig.1 Structure of the multi-server queuing system

Here the following notation is used:

τ_S^1, τ_S^2 are the servicing times of a job for private and leased servers correspondingly,

$\tau_S^3 = \tau_r$ is the fixed waiting (repeating) time for a job,

P_S^1 is the probability of that a number of jobs incoming at a moment at node s is less than a number of private servers N_S,

P_S^2 is the probability of that a number of jobs incoming at a moment at node s is more than a number of private servers N_S, but less than the sum of leased and private servers $N_S + N_C$,

P_S^3 is the probability of that a number of jobs incoming at a moment at node s is more than total number of servers $N_S + N_C$ (all the servers are occupied),

It's obvious, that $P_S^1 + P_S^2 + P_S^3 = 1$.

Therefore, the weight average servicing time $\widetilde{\tau}_S$ of one job at node s can be defined as follows

$$\widetilde{\tau}_S = \tau_S^1 P_S^1 + \tau_S^2 P_S^2 + \tau_S^3 P_S^3 \tag{3}$$

By analogy, for rate $\widetilde{\lambda}_S$ of jobs, incoming at node s, and for rate $\widetilde{\lambda}_C$ of jobs, incoming at the network center, we have

$$\widetilde{\lambda}_S = P_S^1 \lambda_S, \ \widetilde{\lambda}_C = \sum_S P_S^2 \lambda_S, \tag{4}$$

For Poisson law, the probability of that for time $\widetilde{\tau}$ a number of incoming jobs is no more than 'n' is determined by integrated distribution function

$$F(n, \widetilde{\tau}) = \sum_{k=0}^{n} P_k(\widetilde{\tau}) = e^{-\lambda \widetilde{\tau}} \sum_{k=0}^{n} \frac{(\lambda \widetilde{\tau})^k}{k!}.$$

If λ is the rate of flow process, n is the maximal number of jobs served by the queuing system simultaneously (number of servers), $\widetilde{\tau}$ is the average service time of a job, then

$$P_S^1 = e^{-\lambda \tau_S} \sum_{k=0}^{N_S} \frac{(\lambda \tau_s)^k}{k!}, \ P_S^2 = e^{-\lambda \tau_S} \sum_{k=N_S+1}^{N_S+N_C} \frac{(\lambda \tau_s)^k}{k!}, \ P_S^3 = 1 - P_S^1 - P_S^2.$$

The values $\widetilde{\tau}_S, \widetilde{\lambda}_S$ and $\widetilde{\lambda}_C$, defined from expressions (3) and (4), are used in formulas (1) and (2) by substituting $\widetilde{\tau}_s \to \widetilde{\tau}_f$.

More possibilities for analysis of QN and making of optimal decision are provided by the simulation model [1], which has not any restrictions for dimension, kind of jobs arrival, discipline and time of servicing. Techniques for optimization using simulation of stochastic systems are studied, for example in [2, 8].

Numerical example

To compose the simulation model the program ARENA has been used, which is a integrated system with simple and efficient programming language [7]. Simulation experiment was conducted for the following conditions:

1) Two kinds of the basic model are chosen for simulation
   a) QN consisting one processing node and a network-center node, that corresponds a multi-channel queuing system with two kinds of servers (private and leased). The private and leased servers have different productivity and cost of idle time.
   b) QN consisting two processing nodes and a network-center node. Two simulation experiments were conducted with symmetrical and unsymmetrical parameters of processing nodes.
2) The variable parameters of basic models are the number of parallel servers at each processing node and network-center node. For 4 examined configurations they are N_C =*35, 30, 25, 20* of leased servers and N_S=*5, 10, 15, 20* of private servers respectively.
3) Average rate of arriving is *1* job every *2 tu*, the total number of arriving jobs is *10000*, the observation period is *10031 tu*.
4) The simulation model is realized for:
   a) the Poisson law of jobs arriving and exponential distribution of servicing time (for verification of analytical model);
   b) the Erlang law of the 2nd degree for jobs arriving and for servicing time distribution (for optimization by simulation).
5) Average servicing time for servers are respectively τ_S =*20 tu* for private and τ_C =*60 tu* for leased server.

6) Discipline of servicing is 'First come-First served'. The waiting queue is infinite. Running of each waiting job is repeated with interval $\tau_R = 150$ *tu*.

The simulation provided the following results:

a) Chosen criterion functions are critical to server distribution between processing nodes and network-center node and depend on it in opposite way.
b) Simulation confirms the adequacy of proposed analytical methods.
c) A branch and bound algorithm was used for global optimization by simulation of the complete model.

Conclusion

The problem of resources allocation in QN can be formulated as task of optimization in the queuing system. The objective function of the task depends on the stochastic and deterministic parameters, such as kind of arrival jobs of flow process, distribution law of service time, capacity of waiting queue, number of parallel servers in the queuing systems.

In general case, there is not an analytical solution for formulated task. However, with some admissions about kind of flow process and service time an analytical solution can be obtained for primary evaluation of possible decision. Such analytical result is obtained in the article for multi-channel queuing system with different kinds of servers and unlimited capacity of waiting queue.

More possibilities for analysis of queuing system and making of optimal decision are provided by the simulation approach, which has not any restrictions for dimension, kind of jobs arrival, discipline and time of servicing.

Conducted simulation experiment confirmed that value of objective function is critical to resource allocation in QN. In particular, a number of servers, assigned for each node, have to be proportional to its productivity.

References

1. Azadivar, F. and Lee, Y., - Optimization of discrete variable stochastic systems by computer simulation. Mathematics and Computers in Simulation, 2, (1988), 331-345.
2. Biethan, J. and Nissen, V., - Combinations of simulation and evolutionary algorithms in management science and economics. Annals of Operations Research, 32, (1994), 183-208.
3. Buzacott, J. and Standridge, J., - Modeling and analysis of manufacturing systems: Wiley&Sons, N.Y., 1993.
4. Chee Hock Ng, - Queuing Modelling Fundamentals: John Wiley & Sons, New York, 1997.
5. Dolgui, A., Utilisation de Ψ-transformation pour l'optimisation des paramètres des modèles de simulation. In: Actes de la Première Conférence Francophone de Modélisation et de Simulation (MOSIM'97), Hermès, Paris, 1997, 451-459.
6. Dolgui, A. and Ofitserov, D., - A Stochastic Method for Discrete and Continuous Optimization in Manufacturing Systems. Journal of Intelligent Manufacturing, 5, (1997), 405-413.
7. Kelton, W.D., Sadowski, R.P. and Sadowski, D.A. - Simulation with Arena, McGraw-Hill, N. Y., 1997.
8. Guariso, G., Hitz, M. and Werthner, H., - An integrated simulation and optimization modeling environment for decision support. Decision Support Systems, 1, (1996), 103-117.
9. Hall, R.W., - Queuing methods for service and manufacturing. Prentice Hall, Englewood Cliffs. N. Y, 1991.
10. Zaikin, O. and Ignatiev, V., - A method of analysis of multi-channel queuing models. Izv. AN SSSR (Technical Cybernetics, Academy of Science of USSR), 6, (1973), 86-88.
11. Zaikin, O. and Dolgui, A., Resource assignment in mass demand HighTec assembly manufacturing based on the queuing modelling. In: Proc. International Conference on Industrial Logistics (ICIL'99), St. Petersburg, 1999, University of Southampton Publication, 1999, 200-209.
12. Zaikin, O., Dolgui, A. and Kraszewski P., Simulation Approach to resource allocation in the satellite telecommunication network. In: Proc. 13th European Simulation Multi-conference, Warsaw, 1999, SCS Publication, Delft, 1999, 157-161.

DESIGNING STABILIZATION POLICIES IN AN UNCERTAIN ENVIRONMENT

Reinhard Neck[1] **and Sohbet Karbuz**[2]
[1] Department of Economics, University of Klagenfurt
Universitaetsstrasse 65–67, A-9020 Klagenfurt, Austria
Email: reinhard.neck@uni-klu.ac.at
[2] IEA / OECD 332
9, rue de la fédération, F-75739 Paris Cedex, France
Email: sohbet.karbuz@iea.org

Abstract. Optimal budgetary policies for the period 1995 to 2000 are calculated for Austria within the framework of a problem of quantitative economic policy. An intertemporal objective function is minimized subject to the constraints of a macroeconometric model. Using the optimum control algorithm OPTCON, approximately optimal policies are determined. The sensitivity of optimal policies with respect to the development of the exogenous variables of the model is investigated.

Introduction

For a long time, there has been a consensus in the macroeconomics literature that budgetary policies should not be directed exclusively towards fiscal considerations in a narrow sense. Instead, their consequences for other objectives of economic policy-making (such as full employment, price stability, growth, and the external balance) should be taken into account within the conception of comprehensive fiscal stabilization policies. During the last decade, however, these ideas have been exposed to serious criticism. Among other arguments, it has been conjectured that in the age of globalization, an appropriate design of stabilization policies is extremely difficult or even impossible, at least for a small open economy. Strong international linkages combined with severe uncertainties about the development of the global economy may impede planning for fiscal policy measures which aim at more than mere fulfilment of purely fiscal goals such as the "Maastricht criteria" in the European Monetary Union. Hence the question arises whether budgetary policies can be successfully designed as stabilization policies in a wider sense for a small open economy in the presence of considerable uncertainties about the global environment.

In this paper, these issues are analyzed within a problem of quantitative economic policy, using an optimum control approach. Optimal budgetary policies are determined numerically by minimizing an intertemporal objective function subject to the constraints given by an econometric model. The model, called FINPOL3, is a medium-size macroeconometric model for Austria. Moreover, an objective function for Austrian policy-makers over the years 1995 to 2000 is postulated, which penalizes deviations of objective variables from their desired values. The exogenous variables of the model are forecast over this planning horizon.

The primary focus of this study is the sensitivity of optimal policies with respect to the non-controlled exogenous variables. To answer this question, several optimum control experiments are performed under varying assumptions about the paths of the exogenous variables. The results show that there may be strong systematic dependencies of optimal policies on these paths.

The econometric model FINPOL3

The model FINPOL3 is based on traditional Keynesian macroeconomic theory in the sense of conventional IS-LM/aggregate demand-aggregate supply models. Stochastic behavioral equations for the demand side include a consumption function, an investment function, an import function and an interest-rate equation as a reduced-form money market model. Prices are largely determined by aggregate demand variables. Disequilibrium in the labor market, as measured by the excess of unemployed persons over vacancies, is modeled to depend on the real GDP growth rate and the rate of inflation, embodying both an Okun's law-type relation and a rudimentary Phillips curve. The main objective variables of Austrian economic policies, such as real GDP, the labor market disequilibrium variable (related to the rate of unemployment), the rate of inflation, and the balance of payments and the federal net budget deficit (ratios to GDP), are related directly or indirectly to the fiscal policy instruments used as control variables, namely federal budget expenditures and revenues.

The model, which is dynamic and nonlinear, was estimated first by 3SLS using annual data over the period 1965 to 1994. Data have been obtained from the Austrian Institute of Economic Research (WIFO). The estimates and test statistics together with ex-post simulation results suggest that the model provides a reasonable account of the development of economic variables in the recent past. The estimation results together with the statistical characteristics of the regressions are given in [4].

The optimum control approach

In the theory of quantitative economic policy, optimum control theory has been used in several studies to determine optimal policies for econometric models (e.g., [1], [2]). Here the algorithm OPTCON, developed by Matulka and Neck [3], is used; it determines approximate solutions of deterministic or stochastic optimum control problems with a quadratic objective function and a nonlinear multivariable dynamic model. The objective function is quadratic in the deviations of the state and control variables from their respective desired values. The dynamic system is required to be given in a state space representation.

For the simulation experiments, the planning horizon is chosen as 1995 to 2000. Among the variables whose deviations from desired values are to be penalized, two categories are distinguished: First, there are five "main" objective variables which are of direct political relevance in assessing the performance of the Austrian economy. These are the rate of inflation ($PV\%_t$), the labor market excess supply variable (UN_t) as a measure for involuntary unemployment, the rate of growth of real GDP ($YR\%_t$), and the current account ($LBR\%_t$) and the federal net budget deficit ($DEF\%_t$), both as percentages of GDP. In all experiments, 2% p.a. is considered as the desired rate of inflation ($PV\%_t$), 4% p.a. as the desired real growth rate ($YR\%_t$), and the desired levels for labor market excess supply (UN_t) and the current account ($LBR\%_t$) are set equal to zero. For the deficit variable, we assume that the aim is to consolidate the federal budget deficit gradually such that the desired value of $DEF\%_t$ is reduced by 0.5 percentage points each year, from the historical value of 4.67% in 1994 down to 1.67% in 2000.

Second, a category of "minor" objective variables is introduced. These include real private consumption, real private investment, real imports of goods and services, the nominal rate of interest, real GDP, real total aggregate demand, the domestic price level, the price level of public consumption, nominal public consumption, and nominal public-sector net tax revenues, as well as the policy instrument (control) variables federal budget net expenditures (NEX_t) and federal budget tax receipts (BIN_t). We take 1994 historical values of these "minor" objective variables (except for the interest rate) to be given and postulate desired growth rates of 4% p.a. for all real variables, desired growth rates of 2% p.a. for the price level variables, and desired growth rates of 6% p.a. for the nominal variables. The rate of interest has a desired constant value of 6 for all periods.

In the weight matrix of the objective function, all off-diagonal elements are set equal to zero, and the main diagonal elements are given weights of 10 for the "main" objective variables and of 1 for the "minor" objective variables. The state variables that are not mentioned above get weights of zero, thus being regarded as irrelevant to the hypothetical policy-maker. The weight matrix is assumed to be constant over time.

The algorithm OPTCON assumes the values of the non-controlled exogenous variables to be known in advance for all time periods of the planning horizon. For a simulation over a future planning horizon, projections (forecasts) of the exogenous (controlled and non-controlled) variables are needed. Here we use extrapolations of these variables calculated from linear stochastic time series models of the ARMA (mixed autoregressive-moving average process) type. The forecasts from these time series models imply an average growth rate of 5.5% for the fiscal policy variables. The extrapolation implies that the federal budget deficit grows moderately from 99.5 billions ATS in 1995 to 118.8 billions ATS in 2000, which is an optimistic forecast. The development of the foreign sector variables is also optimistic: the import price level grows by less than 1% p.a., and real exports of goods and services grow by 4% p.a. on average. Money supply grows by 6.5% p.a. on average.

Projected versus optimal budgetary policies

As a first step, the model was simulated over the years 1995 to 2000 using the extrapolations of all exogenous variables from the time series models as input. This amounts to a dynamic forecast of the endogenous variables of the model. Next, several optimization experiments were performed. Here different time paths of the non-controlled exogenous variables are used as inputs, being assumed to be known for certain, but the values of the policy instruments are determined endogenously as (approximately) optimal under the assumed objective function. All experiments considered here are deterministic ones.

The projection scenario using the values of the non-controlled exogenous variables from the time series models forecasts a fairly constant inflation rate ($PV\%_t$) of about 2.5%, relatively high growth of real GDP ($YR\%_t$), diminishing involuntary unemployment (UN_t) and a federal budget deficit growing slower than GDP ($DEF\%_t$), implying only minor growth of the debt-to-GDP ratio. Some fluctuations occur along the projected growth path of the Austrian economy, especially in real income and unemployment. The current account exhibits a growing deficit. Details can be found in [4].

Several optimization experiments with the model and the assumed intertemporal objective function were conducted to examine the sensitivity of optimal budgetary policies with respect to the time paths of the non-controlled exogenous variables. As a benchmark, for experiment 1 we use again the values of those variables from the ARMA models. Results for the instrument variables and the "main" objective variables from experiment 1 are given in Table 1. Optimal budgetary policies are more countercyclical than projected ones and imply

smoother time paths of the endogenous variables of the model. In particular, for those years where the projection forecasts lower than desired growth, federal budget expenditures (NEX_t) are higher than in the projection and federal budget revenues (BIN_t) are lower. The reverse is true for the year 1996, where the projection scenario implies higher growth of real GDP. To compensate for expansionary budget expenditures, budget revenues increase faster over the entire planning horizon. The overall effects on policy objective variables are favorable: Unemployment, though initially higher, comes down to lower values than in the projection, the current account deficit and the budget deficit are eventually lower, real GDP growth is higher, and the rate of inflation is virtually unaffected by this relatively expansionary fiscal policy design.

Table 1: Optimal values of instruments and "main" objectives (experiment 1)

year	NEX_t	BIN_t	$PV\%_t$	UN_t	$YR\%_t$	$LBR\%_t$	$DEF\%_t$
1995	789.285	672.909	2.422	4.879	3.813	–1.238	4.851
1996	824.079	721.674	2.572	4.022	4.548	–1.027	3.960
1997	881.422	766.342	2.490	3.861	3.360	–1.620	4.175
1998	929.493	814.704	2.553	3.509	3.939	–1.786	3.882
1999	986.307	861.199	2.550	3.356	3.665	–2.129	3.954
2000	1048.852	905.979	2.550	3.293	3.536	–2.519	4.223

Optimal policies under different assumptions about exogenous variables

The optimization experiment described in the previous section shows that the performance of the Austrian economy can be improved by countercyclical budgetary policies. When such a result is to be communicated to actual policy-makers, its robustness with respect to the underlying assumptions should be checked first. To do so, we have conducted several alternative optimum control experiments, using alternative values of the parameters of the objective function. The results, which are reported elsewhere, show that in most cases optimal policies are quite similar to those of the previous experiment. On the other hand, one might expect optimal policies to depend upon the assumptions made about future developments of the world economy as expressed by the forecasts of the non-controlled exogenous variables of our model, in particular real exports (XR_t) and import prices (PM_t). It is well known from theoretical and empirical studies that macroeconomic developments in a small open economy like Austria are crucially influenced by global business cycles; hence, optimal national budgetary policies should depend on them, too. Moreover, the forecasts of the exogenous variables obtained from the time series models are rather unreliable, providing another reason for exploring the influence of alternative assumptions about global developments upon optimal Austrian budgetary policies.

Among the exogenous variables of the model FINPOL3, import prices (PM_t) and real exports of goods and services (XR_t) are most interesting. Therefore, we use arbitrary annual growth rates for import prices and real exports to construct alternative global scenarios. In experiment 2, we assume constant XR_t and PM_t growing by 4% annually over the time horizon considered (1995 to 2000). This means very high growth of import prices and zero growth of real exports and can be characterized as an extremely "pessimistic" scenario. The same values as in the previous simulation were used for the other exogenous variables. The results are shown for the optimization in Table 2.

Table 2: Optimal values of instruments and "main" objectives, "pessimistic" scenario (experiment 2)

year	NEX_t	BIN_t	$PV\%_t$	UN_t	$YR\%_t$	$LBR\%_t$	$DEF\%_t$
1995	802.984	563.634	3.183	5.619	1.562	–2.931	10.208
1996	873.032	586.194	3.377	4.762	3.694	–3.975	11.450
1997	953.198	614.260	3.439	4.189	3.513	–5.292	12.667
1998	1048.402	660.169	3.469	3.942	3.045	–6.662	13.633
1999	1154.548	737.332	3.435	4.105	2.164	–7.893	13.887
2000	1253.884	846.286	3.302	4.808	0.713	–8.773	13.064

Comparing these results with those of Table 1, we see considerable differences. In the projection (with unchanged budgetary policies), the lower growth rate of real exports implies a slowdown of Austrian economic activity. Table 2 shows that the optimal reaction of Austrian budgetary policies (given the postulated objective function) is highly expansionary: Federal expenditures (NEX_t) are always considerably higher, federal revenues (BIN_t) are always considerably lower than in the projection. This is also true when compared with the optimal policies of the previous (more "optimistic") scenario of experiment 1 (Table 1). The result of these policies is a much better performance than in the projection with respect to growth and unemployment at the expense of higher inflation, deficits of the current account and considerably higher budget deficits (up to 408 billions ATS or 13% of GDP in 2000). Especially these budget deficits will not be sustainable in the long run, but this aspect

cannot be taken into account in a short-run Keynesian model like FINPOL3. Nevertheless, the performance of the Austrian economy will be much worse in this scenario than in the one of experiment 1, even in spite of the extremely expansionary fiscal policies applied, as can be seen by comparing the results in Tables 1 and 2.

Quite different outcomes are obtained if we assume an extremely "optimistic" scenario. For this purpose, we assume import prices (PM_t) and real exports (XR_t) to grow by 6% and –2%, respectively, per year over the entire time horizon in experiment 3. Given experiences of the recent past, such a development is highly unlikely in the near future; it is simulated here just for the purpose of pointing out the effects of a very "optimistic" view about a sustained global boom with falling import prices on optimal policies for Austria. The main results are shown in Table 3. Under unchanged budgetary policies, this scenario leads to an overheating of the Austrian economy. In this case, the optimal reaction of budgetary policies consists in a quick reduction of budget deficits, leading eventually to a balanced federal budget in the second half of the planning period. This is brought about both by lower federal expenditures and higher taxes, as compared to the scenario of experiment 1. These relatively restrictive policies reduce real growth and inflation, as compared to the corresponding projection. When the results are compared to those of optimal fiscal policies in experiment 1 (Table 1), however, the performance of all variables is improved. Again, the demand-side effects dominate, as has to be expected from the structure of our Keynesian model FINPOL3. Budgetary policies are an effective instrument in this context, but their optimal design is crucially dependent on the assumed developments in the world economy.

Table 3: Optimal values of instruments and "main" objectives, "optimistic" scenario (experiment 3)

year	NEX_t	BIN_t	$PV\%_t$	UN_t	$YR\%_t$	$LBR\%_t$	$DEF\%_t$
1995	801.382	701.407	1.586	4.972	3.935	–1.023	4.166
1996	828.778	770.996	1.621	4.265	4.510	–0.358	2.239
1997	855.858	834.398	1.612	3.849	4.241	–0.002	0.775
1998	883.446	887.159	1.636	3.555	4.196	0.229	–0.125
1999	912.028	923.102	1.720	3.205	4.506	0.347	–0.346
2000	947.210	938.021	1.892	2.635	5.238	0.255	0.264

Conclusions

In this paper, we have used a medium-size macroeconometric model of the Austrian economy to calculate optimal budgetary policies for the years 1995 to 2000 for a given objective function. If we compare the results of the optimization runs to simulations with extrapolations of policy instruments used as inputs, optimal policies turn out to be more countercyclical and to dampen the amplitude of business cycle fluctuations. If this is in fact a goal of economic policy-making, using an optimum control approach within a framework of quantitative economic policy might be recommended to political decision-makers and their advisers as an instrument to generate insights into possibilities for improving policy-making. However, alternative assumptions about the development of non-controlled exogenous variables reflecting global developments have been shown to change optimal budgetary policies considerably. This implies that the reliability of policy recommendations depends strongly on the quality of the forecasts for the exogenous variables, and great caution is required in interpreting results from optimization experiments for policy purposes. Moreover, comparisons between model results of optimal stabilization policies under widely differing assumptions about exogenous variables show that the main determinant of the macroeconomic performance of a small open economy like the Austrian one is not the design of budgetary policies but the development of the global economy, which constrains the scope of national policy-makers' sovereignty and effectiveness considerably.

Acknowledgement

Financial support from the "Jubilaeumsfonds der Oesterreichischen Nationalbank" (project no. 6917) and from the Ludwig Boltzmann Institute for Economic Analysis is gratefully acknowledged. The views expressed are not necessarily those of the IEA/OECD.

References

1. Chow, G. C., Econometric Analysis by Control Methods. Wiley, New York, 1981.
2. Kendrick, D., Stochastic Control for Economic Models. McGraw Hill, New York, 1981.
3. Matulka, J. and Neck, R., OPTCON: An algorithm for the optimal control of nonlinear stochastic models. Annals of Operations Research, 37 (1992), 375 – 401.
4. Neck, R. and Karbuz, S., Optimal control of fiscal policies for Austria: applications of a stochastic control algorithm. Nonlinear Analysis, Theory, Methods & Applications, 30 (1997), 1051 – 1061.

MODELLING OF ORGANIZATIONAL DECISION-MAKING SYSTEMS AND DECISION PROCESSES

Dr.László Cserny
Dunaújváros Polytechnic of Miskolc University, Institute of Informatics
H-2400 Dunaújváros, Táncsics M.u.1/a, Hungary
e-mail: cserny@mail.poliod.hu

Abstract. The investigation and modelling of organizations as well as that of their decision-making systems and decision processes are nowadays problems of great importance. Studying these matters is especially necessary before the implementation of any decision support system(DSS) or executive information system(EIS). In my paper, I will deal with these subjects with reference to the results of [1] and to [2,3] as theoretical background.

Elementary decision-making systems and state space

The base of our investigation and model is a model of decision-making systems consisting of a single activity(Figure 1a).

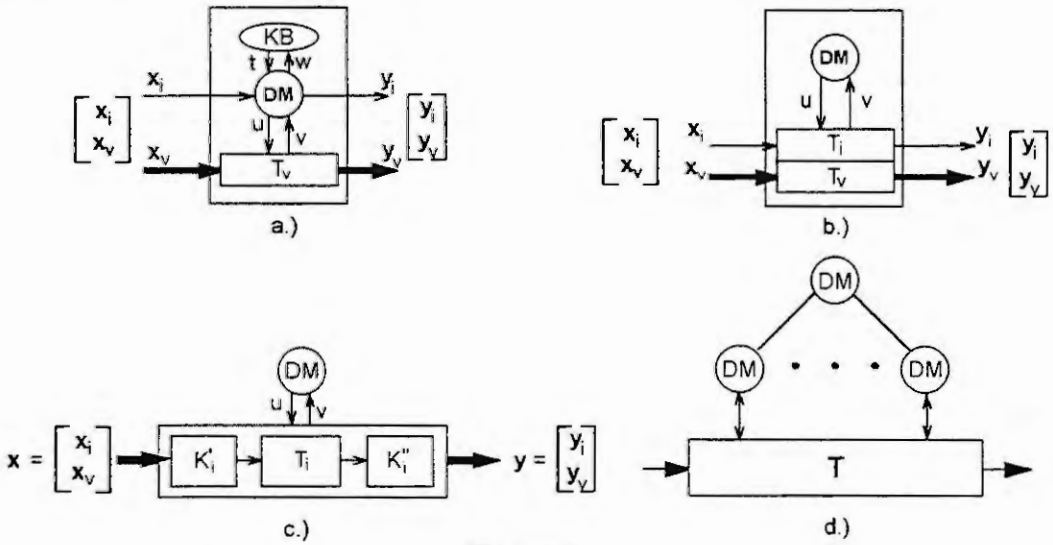


Figure 1

Any changes of the system's states will be analysed in the state space of the input(x) and output(y) state variables and of the result of a goal function z given by (1), that is in the form (2) of the state and of the state space given by (3).

$$\mathbf{z} = \mathbf{f}(\mathbf{x},\mathbf{y}) = (f_1(\mathbf{x},\mathbf{y}), f_2(\mathbf{x},\mathbf{y}),\ldots, f_l(\mathbf{x},\mathbf{y})) = (z_1, z_2,\ldots z_l) \tag{1}$$

$$\mathbf{s} = (\mathbf{x},\mathbf{y},\mathbf{z}) = (\mathbf{x},\mathbf{y},\mathbf{f}(\mathbf{x},\mathbf{y})) = (s_1, s_2,\ldots, s_n,\ldots, s_{n+l}) \tag{2}$$

$$S = X \times Y \times Z \tag{3}$$

Thus, considering Figure 1a and not treating the connections of the knowledge base KB, the decision-maker(DM)'s state space can be written in the following form:

$$D = X_i \times V \times Y_i \times U \tag{4}$$

The elementary decision-making systems can be decomposed into two layers (Figure 1b):

- the layer T_v of the execution of the activity and
- the layer T_i of controlling, of information processing.

Organizational decision-making systems

We look at the organizational decision-making systems as the hierarchical system of the elementary decision-making systems embedding one into another.

The level of execution. The decomposition of the transformation level of the system can be seen in the Figure 1c, where K'_i is the input unit that can summarise the incoming resources and transfer to the activity T_i that is followed by the unit K''_i distributing the output of T_i to the other parts of the process. On the basis of the transformation process (Figure 2) which can be converted into the structure in Figure 3 we may write that

$$T : E' \times X \rightarrow E'' \times Y \tag{5}$$

$$K : E'' \rightarrow E' \tag{6}$$

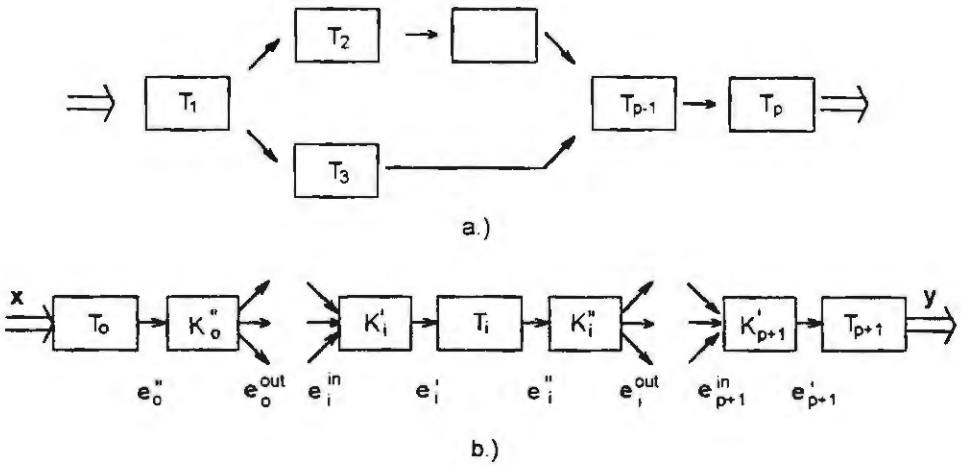


Figure 2

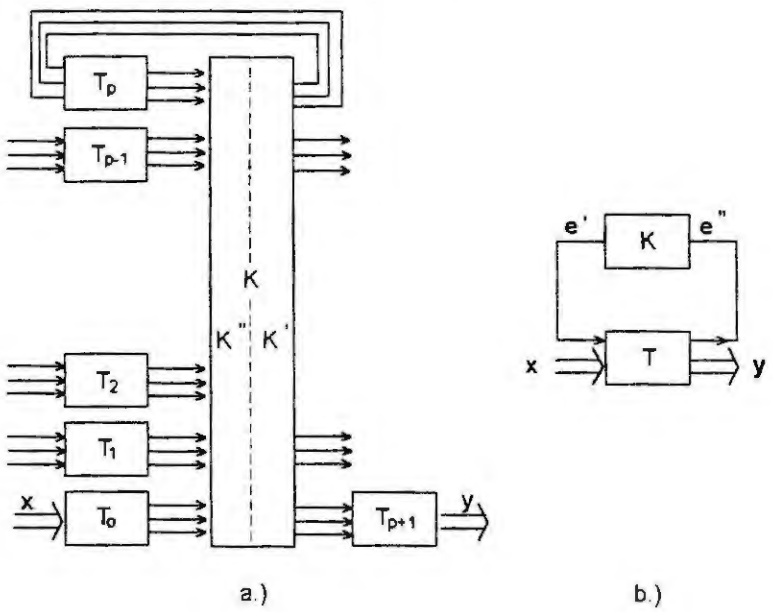


Figure 3

The level of controlling, of information processing. The connections of the elementary systems and those of the systems embedded into one another considered as the structure of the whole system can be seen in Figure 4 and Figure 5, respectively.

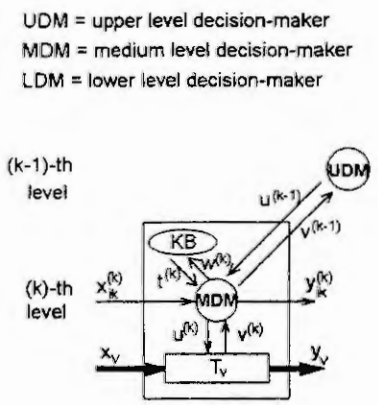


Figure 4

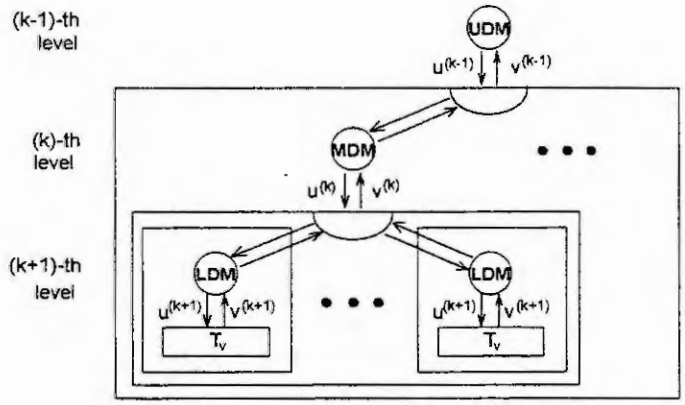


Figure 5

Using the notation of the figures the transformation of X into Y can be expressed in the form of (7),

$$\begin{aligned} &T^{(k)}: X^{(k)} \to Y^{(k)} \qquad \textit{that is} \\ &\quad T^{(k)}: U^{(k-1)} \times X_{ik}^{(k)} \times X_v^{(k)} \to V^{(k-1)} \times Y_{ik}^{(k)} \times Y_v^{(k)} \qquad k = 2,\ldots,r \\ &\quad \textit{and at the uppermost level} \\ &\quad T^{(k)}: X_{ik}^{(k)} \times X_v^{(k)} \to Y_{ik}^{(k)} \times Y_v^{(k)} \qquad k = 1 \end{aligned} \tag{7}$$

where

$U^{(k-1)}$ is the set of statements, of commands from the level (k-1),

$X_{ik}^{(k)}, Y_{ik}^{(k)}$ are the sets of input and output information from and to the environment on the level (k),

$X_v^{(k)} = X_{ik}^{(k+1)} \times X_v^{(k+1)}$ is the set of input of the execution level (k),

$Y_v^{(k)} = Y_{ik}^{(k+1)} \times Y_v^{(k+1)}$ is the set of output of the execution level (k),

$V^{(k-1)}$ is the set of reports to the level (k-1).

The decision-maker's input and output are given by (8) and (9).

$$X_D^{(k)} = U^{(k-1)} \times X_{ik}^{(k)} \times V^{(k)}, \qquad k = 2,\ldots,r \quad \textit{and} \quad U^{(k-1)} = \varnothing \quad \textit{if} \quad k = 1 \tag{8}$$

$$Y_D^{(k)} = V^{(k-1)} \times Y_{ik}^{(k)} \times U^{(k)}, \qquad k = 2,\ldots,r \quad \textit{and} \quad V^{(k-1)} = \varnothing \quad \textit{if} \quad k = 1 \tag{9}$$

Considering the decision-maker's input and output given by (8), (9) and the result set Z_D of the goal function we can determine the decision-maker's state space in the form of (10).

$$D^{(k)} = X_D^{(k)} \times Y_D^{(k)} \times Z_D^{(k)} \qquad \textit{for} \quad \forall k \tag{10}$$

Controlling and co-ordinating of decision-making systems. The decision-makers ought to determine their decision strategies in such a way that if

$$\Phi = \left\{ \varphi \middle| \varphi = \left\langle \gamma_1, \gamma_2, \ldots, \gamma_i, \ldots, \gamma_p \right\rangle, \quad \textit{for} \quad \forall i, \quad \gamma_i : A_i \to A_i^* = \{a_i^*\} \right\} \tag{11}$$

is the set of the decision-maker's strategies, where

A_i is the set of courses of acting belonging to the decision γ_i,

A_i^* is the set of optimal (satisficing) courses of acting resulted by γ_i,

then $g[\varphi] \to \min$ should be realised for some strategy $\varphi \in \Phi$ and goal-functional $g[.]$. (Here, the set of the courses of acting A is assigned to the transitions of the decision state space D, that is there is a function h so, that $h: D \times D \to A$.)

If $\varphi^o \in \Phi^o$ is a UDM's strategy then the objective of the LDM is to find a strategy $\varphi \in \Phi_L \subseteq \Phi$ so that $g\left[\varphi \mid \varphi^o\right] \to \min$, that is $\left\| g[\varphi] - g\left[\varphi \mid \varphi^o\right] \right\| \to \min$ should be, where Φ_L is the set of the LDM's possible strategies belonging to the UDM's strategy φ^o.

Every decision-maker($i = 1,2,...,t$) has a subset $R_i(\varphi^o)$ of his possible strategies that consists of the best strategies of his own responding to some UDM's strategy $\varphi^o \in \Phi^o$. We note these sets(given by the formula (12)) as the sets of the LDMs' **rational strategies** or in the case of a group of subordinate LDMs, as the set(expressed in the form (13)) of **collective rational strategies** respectively.

$$R_i(\varphi^o) = \left\{ \varphi^* \mid g_i(\varphi^* \mid \varphi^o) \le g_i(\varphi \mid \varphi^o),\ \varphi, \varphi^* \in \Phi_{iL} \subseteq \Phi_i \right\} \qquad \forall i - re \tag{12}$$

$$R(\varphi^o) = R_1(\varphi^o) \times R_2(\varphi^o) \times ... \times R_t(\varphi^o) \tag{13}$$

We can determine for every co-operation mode of the DMs (e.g. the situation of independent DMs, the co-operation in the situation of Pareto- or Nash-equilibrium) its own set of collective rational strategies. In this case the UDM ought to choose such strategy $\varphi^{o*} \in \Phi$ that the LDMs' collective rational strategy belonging to this chosen UDM strategy results in better solution than any other case of choice $\varphi^o \in \Phi^o$ from the set Φ^o of the UDM's (rational) possible strategies, that is

$$g^o(\varphi^{o*} \mid \underline{\varphi}^*) \le g^o(\varphi^o \mid \underline{\varphi}) \qquad \varphi^{o*}, \varphi^o \in \Phi^o \qquad \acute{e}s \qquad \underline{\varphi}^* \in R(\varphi^{o*}), \underline{\varphi} \in R(\varphi^o) \tag{14}$$

Summary

The study is a short summary of the detailed mathematical description of the organization and of its decision-making system. On the basis of the results of such analysis we will be able to determine the decision points of the organization(the points where decisions taken by the assigned decision makers), the relations and links of these points, the type of connections, the decisions and their attributes. We will be able to learn and understand the structure and the functioning of the organizational decision-making system before the development and implementation of a DSS or EIS, or any other intelligent information system.

References

[1] *Cserny,L.*: The Analysis of Decision-Making Systems, in: Sydow,A.- Tzafestas,S.G.-Vichnevetsky,R.(eds.): Systems Analysis and Simulation 1988 I., Akademie-Verlag, Berlin, 53-58, 1988 (3rd International Symposium on Systems Analysis and Simulation, Berlin, 1988)

[2] *Kickert,W.J.M.*: Organization of Decision-Making. A Systems Theoretical Approach, North-Holland, Amsterdam-New York-Oxford, 1980

[3] *Mesarovic,M.D.-Macko,D.-Takahara,Y.*: Theory of Hierarchical, Multilevel Systems, Academic Press, New York-London, 1970

THE DYNAMIC INTERACTION BETWEEN ECONOMY AND ECOLOGY

Cooperation, Stability and Sustainability for a Dynamic-Game Model of Resource Conflicts

J. Scheffran
Fachbereich Mathematik, Technical University Darmstadt
Schlossgartenstraße 7, D–64289 Darmstadt

Abstract. The interaction between economic and ecologic dynamic systems is analyzed with a multi-player dynamic game, where each player invests and allocates available capital to the production or consumption of natural resources and goods and evaluates the outcome of all players' actions as well as the reactions of the ecosystem. During the course of repeated actions a dynamic learning process evolves such that the players adapt the amount of investment and the direction of its allocation according to their action preferences. For certain stability conditions of the interaction matrix the players can form coalitions. Cooperation and negotiation on the amount and the distribution of investment could lead to more sustainable use of natural resources. The dynamic coalition-formation process corresponds to a self-organized transition from unilateral action (Nash equilibria) to multilateral cooperation (Pareto optima). With increasing number of players the complexity-control tradeoff can determine limits for stable coalition size. As an example for resource conflicts the exploitation of fish resources is simulated, for different kinds of fish populations and different players harvesting fish.

Introduction

Social structures and processes are based on the decisions, actions and perceptions of agents (also called players) who – in the context of their values, their available means and their freedom of action – pursue goals and interact with other agents. Phase transitions can occur if changes in the value and preference structures on the individual "micro level" directly influence the societal "macro level". With regard to the goals and the allocation of means, contradictions and conflicts may occur, preventing the successful achievement of goals for some agents. Conflict prevention, avoiding destructive escalation and negative benefit-cost ratios, involves a communication and negotiation process aiming at the mediation between contradictory positions and exploring the possibilities of cooperation.

Models of social interaction on the one hand apply the theory of dynamic systems, analyzing equilibria and stability, chaos and self-organization. On the other hand they use game theory, dealing with rational choice between options in a reactive social environment. Between *agent models* and *system models* there still is a methodological gap. In recent years a variety of models are being developed to close the gap, including differential games and evolutionary games as well as computer models of artificial societies.

Most appropriate to analyze the evolution of cooperation and the complexity of social interaction are dynamic game models in which players at the same time adapt their behavior to and shape their natural and social environment, according to their own capabilities, incentives and preferences. The agent's behavior can be represented by the agent triangle which describes the interaction between a *system state* X, its perceived *value* $V(X)$ for the agent compared to a *value target* V^* (objective, goal), and the amount of *invested capital* C (costs), allocated according to *preferences* p to actions changing the system state. The variables are linked by the processes of *observation* of the system, the *decision* on the amount of investment and its preference and the *action* in which these are actually applied. If several agents repeatedly behave in the same way in a systems environment, a social interaction dynamics evolves which can be controlled by all agents according to their action power.

Such an approach is appropriate in economic theory, but in the environmental sciences as well, where systemic approaches (from ecology) and agent-based approaches (economic exchange, political decisions, social relations) are directly linked. An unresolved problem is how the individual behavior can be adapted to the ecological necessities. Under which conditions can a self-organized, collective learning process to sustainable modes of behavior emerge? To examine this question, in the following a model is presented and applied to resource conflicts and its cooperative resolution (based on [8] and 9]), using the competition on scarce fish resources as an example for computer simulation.

The dynamic game of value-capital system interaction

The framework model describes the dynamic game between players Pi ($i = 1, ..., n$) who invest capital C_i to influence system variables x^k according to their allocation preferences p_i^k and evaluate the outcomes

according to a value function V_i. The output vector V is a function of the input vector C, the system state x and the preference matrix p (see Figure 1):

$$V = f(x, p, C) \tag{1}$$

$x = (x^1, ..., x^k)$: vector of system variables (resources) x^k $(k = 1, ..., m)$
$C = (C_1, ..., C_n)$: input vector of invested capital (costs) for players Pi $(i = 1, ..., n)$, where $0 \leq C_i \leq C_i^+$ is the feasible capital range
$V = (V_1, ..., V_n)$: output vector of values for players Pi $(i = 1, ..., n)$
$p : n \times m$ matrix of allocation preferences for investment C_i to variable x^k $(0 \leq p_i^k \leq 1, \sum_k p_i^k = 1)$
$f = (f_1, ..., f_n)$: vector of transition functions f_i between cost input C and value output V

The control problem in this dynamic game concerns the following question: for a given state (x, p, C, V), under which conditions can the agents – by controlling their investment flow $0 \leq C_i \leq C_i^+$ and their allocation preferences $0 \leq p_i^k \leq 1$ – achieve their value targets $V_i = V_i^*$? A solution is possible if player i can resolve equation (1) for the required input $\tilde{C}_i$ or the required preference vector $\tilde{p}_i$ such that

$$\begin{aligned} \tilde{C}_i &= g_i^C(C_{-i}, V_i^*) \\ \tilde{p}_i &= g_i^p(p_{-i}, V_i^*) \end{aligned} \tag{2}$$

where C_{-i} is the vector of the costs of all other players, except C_i, and $p_{\cdot}$ the matrix of the preferences of all players, except column i. Individual solutions can be found by player i, if his individual target sets $\tilde{C}_i = \{\tilde{C}_i(C_{-i} \mid V_i(C) = V_i^*\}$ are not empty and intersect with the feasible set $\mathcal{C} = \{C \mid 0 \leq C_i \leq C_i^+, i = 1, ..., n\}$: $\tilde{C}_i \cap \mathcal{C} \neq \emptyset$, which depends on the form of the inverse functions g_i^C. A similar condition holds for the preferences. If target sets for player Pi exist in the feasible cost and preference space, then Pi can move towards his cost and preference target sets according to the adaptative learning algorithm

$$\begin{aligned} \Delta C_i &= \alpha_i^C(\tilde{C}_i - C_i) \\ \Delta p_i &= \alpha_i^p(\tilde{p}_i - p_i) \end{aligned} \tag{3}$$

where $p = (p_i^1, ..., p_i^m)^T$, and α_i^C , α_i^k are the reaction parameters, measuring the required speed of approaching the target sets. Δ represents both differential changes for continuous time as well as differences for discrete time. For $\alpha_i^C = \alpha_i^k = 1$ the gap is closed in one time-step, and the agents jump back and forth between their target sets (a similar approach has been applied to cost space in [5]).

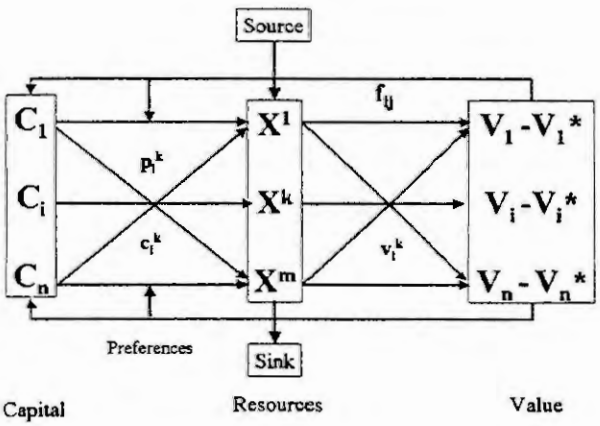


Figure 1: The interaction cycle between costs, values, resources and preferences for multiple players.

All players together can achieve their value targets only if $\hat{\mathcal{C}} = \{C \mid V(C) = V^*\} \neq \emptyset$ which defines the *balance of costs*. All players belonging to such a collective target set can form a *cost coalition* of players with compatible value targets. New players could become part of a coalition either by adapting their preference in a way that makes their actions compatible with the other players, or by changing their value functions and value targets. If the corresponding reaction curves are drawn in preference space for all agents, they could under certain conditions enclose an area of reduced balance costs for

all agents. Moving into this *cooperation channel* requires coordinated action and negotiations on the distribution of the saved costs to which cooperative game theory can be applied (see [4], [6]). A similar analysis holds for *preference coalitions*. A special case of value targets are the optimal values, at which player Pi cannot further improve his individual value for given costs C_{-i} and preferences p_i of the other players. Necessary conditions for optimal values are

$$\frac{\partial V_i}{\partial C_i} = 0, \quad \frac{\partial V_i}{\partial p_i^k} = 0.$$

The solutions C_i^* and p_i^* $(i = 1, ..., n)$ represent Nash equilibria and can be used as target sets $\tilde{C}_i$ and $\tilde{p}_i$, resulting in an individually optimizing dynamics.

Coalition stability and complexity

To assess the stability of the value-cost interaction, we determine the impact of investment changes ΔC, preference changes Δp and system changes Δx on value changes ΔV:

$$\Delta V = F^C \cdot \Delta C + F^p \cdot \Delta p + F^x \cdot \Delta x$$

where F^C, F^p, F^x are the respective interaction matrices for C, p, x. For linear problems or sufficiently small (differential) changes $\Delta C, \Delta p, \Delta x$ the elements of the interaction matrices are given by

$$f_{ij}^C \equiv \frac{\partial f_i}{\partial C_j}, \qquad f_{ij}^k \equiv \frac{\partial f_i}{\partial p_j^k}, \qquad f_{ik}^x \equiv \frac{\partial f_i}{\partial x_k}.$$

In the following we discuss the stability of the value-cost interaction. Using the new variables $z_i \equiv \Delta V_i$ and $y_i \equiv \Delta C_i$, one obtains $z_i = \sum_j f_{ij} \cdot y_j = z_i^* > 0$ where f_{ij} are the elements of the interaction matrix F^C and z_i^* is a required value change ΔY_i^*. Solving this equation for y_i yields a target line in cost space

$$\tilde{y}_i = \frac{z_i^* - \sum_{i \neq j} f_{ij} y_j}{f_{ii}}. \tag{4}$$

Again the dynamic learning algorithm can be applied:

$$\Delta y_i = \alpha_i(\tilde{y}_i - y_i) = \frac{\alpha_i}{f_{ii}}(z_i^* - \sum_{i=1}^{n} f_{ij} y_j) = \frac{\alpha_i}{f_{ii}}(z_i^* - z_i).$$

With $\alpha_i \equiv \alpha$ and $f_{ii} = 1$ for all i (this can be achieved by redefined variables $z_i' = z_i / f_{ii}$ and $f_{ij}' = f_{ij} / f_{ii}$) we obtain the dynamic system $\Delta y = \alpha(z^* - Fy)$. The fixed point in which all players achieve their required value changes can be derived from $Fy = z^*$, for an invertible matrix F given by $\hat{y} = F^{-1} z^*$. According to Cramer's rule the coordinates are $\hat{y}_i = \det F_i^z / \det F$ where F_i^z is the adjoint matrix of F, with column i replaced by target vector z^*. Around this fixed point the dynamics evolves according to $\Delta \bar{y} = -\alpha F \bar{y}$ for $\bar{y} = y - \hat{y}$. The fixed point is stable if the interaction matrix F is stable (for negative Eigenvalues, Hurwitz conditions), corresponding to a stable coalition. Depending on the structure of F various modes of behavior can occur in cost space, including asymptotic stability and instability as well as periodic oscillations and chaotic fluctuations. Since F depends on the system vector x and preference matrix p, changes Δx and Δp have an impact on the dynamics in cost space.

It should be noted that a stable coalition can result from individual action if each player moves his target costs to his target values and the repeated action-reaction mechanism leads to the cost balance at which all players are satisfied. With an increasing number of players the number of eigenvalues of the interaction matrix F increases and therefore the chance that some are positive and the coalition would become unstable. This relationship has been extensively discussed in ecology and population dynamics as the "complexity-stability" tradeoff (see [7], [3]) The results from this debate can be extended to the interaction of individuals and coalitions instead of populations. During social evolution stable coalitions survive while the unstable ones vanish.

If the players want to optimize their collective values $V = \sum_i V_i$ and costs $\hat{C} = \sum_i \hat{C}_i$, the optimum conditions $dV = 0$ and $d\hat{C} = 0$ allow the players to determine a Pareto optimum for which no player can improve without disadvantages for some other players.

The complete dynamic interaction model

Putting all dynamic equations for values V_i, costs C_i, system variables x_i^k and preferences p_i^k together, the following dynamic system is obtained (see Figure 2):

$$\begin{aligned} V_i &= f_i(x,p,C) \\ \Delta C_i &= \alpha_i^C(\tilde{C}_i(V_i^*) - C_i) \\ \Delta p_i^k &= \alpha_i^k(\tilde{p}_i^k(V_i^*) - p_i^k) \\ \Delta x^k &= f^k(x) \pm \sum_j \frac{p_j^k}{c_j^k} C_j \end{aligned} \tag{5}$$

where f^k is the uncontrolled dynamic system for variable x^k, effected by investment C_j in a positive or negative way, with unit costs c_j^k. For a given initial parameter set, defined by transition functions f_i, target values V_i^*, reaction parameters α_i^C and α_i^k and initial values for C_i, x^k, and p_i^k the dynamic evolution of the multi-player interaction can be simulated on a computer. Instead of a fixed adaptation mechanism ΔC_i and Δp_i^k, either of these variables could be free control variables in a dynamic game.

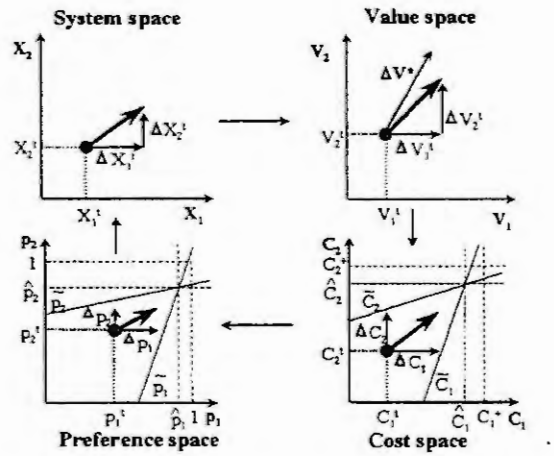


Figure 2: The interaction between the four spaces of the value-cost interaction model.

Competition in resource management

The general methodology is applied to the management of m natural resource stocks x^k which grow with a *regeneration function* $G(x^k)$, change according to an *interaction function* G^{kl} and are *harvested* by n economic players Pi at a time-dependent rate $h_i^k(t)$. This leads to the resource dynamics

$$\Delta x^k = G^k(x^k) + \sum_{l=1}^{m} G^{kl}(x^k, x^l) - \sum_{i=1}^{n} h_i^k \tag{6}$$

In the following the logistic growth function $G^k(x^k) = r^k x^k(1 - x^k/K^k)$ is used, with growth rate r^k and carrying capacity K^k. The harvest $h_i^k = \gamma_i^k \cdot x^k \cdot p_i^k \cdot C_i$ is related to investment C_i, where γ_i^k is the harvest efficiency per cost unit and resource unit of stock x^k. Resource unit costs $c_i^k = \frac{1}{x^k \gamma_i^k}$ approach infinity for resource extinction. p_i^k is the fraction of the efforts of player i allocated to resource x^k ($\sum_k p_i^k = 1$). The net values of the players are

$$V_i = f_i(x,p,C) = \sum_k q^k h^k - C_i = (u_i - \sum_j v_{ij} C_j) C_i \tag{7}$$

with $q^k = a^k - b^k \sum_j h_j^k = a^k - b^k \sum_j \frac{p_j^k}{c_j^k} C_j$: price per unit of harvested resource $h^k = \sum_i h_i^k$

$u_i = \sum_k \frac{a^k}{c_i^k} p_i^k - 1, \qquad v_{ij} = \sum_k \frac{b^k}{c_j^k c_i^k} p_j^k p_i^k.$

The interaction matrix F^C and its stability is determined by the partial derivatives

$$f_{ii}^C = \frac{\partial V_i}{\partial C_i} = u_i - \sum_j v_{ij} C_j - v_{ii} C_i, \quad f_{ij}^C = \frac{\partial V_i}{\partial C_j} = -v_{ij} C_j < 0$$

as a function of C and p. $f_{ii}^C = 0$ leads to the optimal values and the related optimal costs

$$C_i^* = \frac{u_i - \sum_{j \neq i} v_{ij} C_j}{2 v_{ii}}$$

which can be used as target costs $\tilde{C}_i$ for the cost adaptation of player Pi. In the same way the conditions $\partial V_i / \partial p_i^k = 0$ lead to the optimizing target preferences of the form $p_i^{k*} = \beta_i - \sum_{j \neq i} \beta_{ij} p_j = \tilde{p}_i^k$, where β_i and β_{ij} are parameters calculated from the value function. Both target lines in cost and preference space are linear. While this adaptive process provides the individually optimizing behavior, the procedure can be easily modified for optimizing the collective value $V = \sum_i V_i$.

Conditions for sustainability

Much of the debate on sustainable development and environmental economics has focused on ecological limits, operational principles of sustainability, optimal use of natural resources and the optimal control of resource management. ([2], [1]) Agent models are important to understand the compatibility of economic and ecologic dynamics. A fundamental question is how the amount resource extraction $X = h$ of a resource (ignoring index k) by various players can be restricted to not exceed a "critical" resource level X^*, defined as the *sustainable level* $X = \sum_i X_i = \sum_i C_i \cdot p_i / c_i \leq X^*$, although all players act independently. Strategies to achieve this are limits on investment C_i, increasing unit costs c_i and prices q (due to taxes) and adapted preferences p_i. *Sufficiency* requires that the value exceeds the player's target value $V_i \geq V_i^*$. Both conditions together lead to $V_i^* \leq V_i(X) \leq V_i(X^*)$. For $V_i^* > V_i(X^*)$ there is no possible solution (incompatibility). A balance between both requirements is achieved for $V_i = V_i^*$. To mutually achieve *sustainable targeting*, the players need to adapt their actions to one another. From optimal control theory it is possible to calculate optimal resource levels and the total efforts $C^k = \sum_i C_i^k$ of each resource k.[10] The analysis can be repeated with combined effort $C_\gamma = \sum_i \gamma_i C_i$ instead of γC, leading to the steady-state $C_\gamma^{k*} = r^k(1 - x^k/K^k)$. If both agents jointly try to aim for this sustainable target cost, the adequate algorithm is

$$\Delta C_\gamma^k = \beta^k (C_\gamma^{k*} - C_\gamma^k) \tag{8}$$

where $k = 1, ..., m$ and β^k is the reaction strength. The share $\gamma_i^k \Delta C_i^k = \varphi_i^k \Delta C_\gamma^k$ of these changes for each of the agents is a variable to control the process. A plausible apprach is $\varphi_i^k = \gamma_i^k C_i^k / C_\gamma^k$, which implies that those with higher efforts increase or decrease their harvest correspondingly.

Simulation of multi-player fishery management

The theoretical approach is applied to simulate the management of two fish populations x^k $(k = 1, 2)$, harvested by players Pi $(i = 1, ..., n)$. In the baseline set both fishes have the same carrying capacity $K^x = K^y = 1000$, the same growth rate $r^x = r^y = 200/K$ and the same initial density $x(0) = y(0) = 500$. For y the initial price $a^y = 2a^x = 2$ and the price elasticity $b^y = 2b^x = 0.001$ are twice as high as for x, the technical efficiency for catching fish $\gamma^y = 0.006$ is higher $(\gamma^x = 0.004)$. We assume no resource interaction $(G^{kl} = 0)$. For the fishing players, whose number varies, symmetric behavior is assumed in the baseline parameter set. The initial allocation preferences $p_i^k = 0.5$ for both fishes are the same, the initial available capital is $C_i^+(0) = 100$ cost units, the initial investment $C_i(0) = 10$ units for all players. The reaction parameters $\alpha^C = \alpha^p = 0.5$ reflect a 50 % reduction of the gap between optimizing target costs $\tilde{C}$ and preferences $\tilde{p}$ and their actual values in the next time step. The maximum investment is 50 % of the available capital which is increased or decreased according to the net profit or net loss V_i.

Model runs are depicted for 30 time-steps, in different diagrams: (a) Resources x and y, and harvest h_i^x, h_i^y (the latter drawn into the negative direction); (b) prices q^x and q^y and unit costs c_i^k (the latter drawn into the negative direction); (c) accumulated capital (in the diagram CCi denotes C_i^+); (d) invested capital C_i; (e) net profit V_i; (f) preference p_i^k. Two cases are simulated (see Figure 3).

Case 1 (individual profit maximization): Six players are competing for the two fish resources, where the initial allocation preferences p_i^y for fish y are randomly selected from the interval $[0, 1]$ (with $p_i^x = 1 - p_i^y$). The catch efficiencies are varied between the minimum of half and the maximum of twice of the baseline values $\gamma^x = 0.004$ and $\gamma^y = 0.006$ such that player 6 is most efficient and player 1 least efficient in harvesting x, while for fish y it is the other way round. Then initiatially harvest increases to high levels (maximum about 200 for y), with y almost being extinguished to levels of less than 10 % of the initial value. The unit costs for catching low-density y reach very high values (more than ten times the initial costs), the price also increases. For x, price and unit cost are lower, while the stabilized resource level is higher. Only players 1 and 6 can achieve considerable growth in capital (leading in efficiency), while the others decline or slightly improve due to instability of too many competing players on the scarce fish market. Allocation preferences show fluctuations in the beginning until y is almost lost and the players have fixed their preferences to either 0 or 1. Switching between preferences only occurs, when y is trying to recover, without chance against the heavy demand of y.

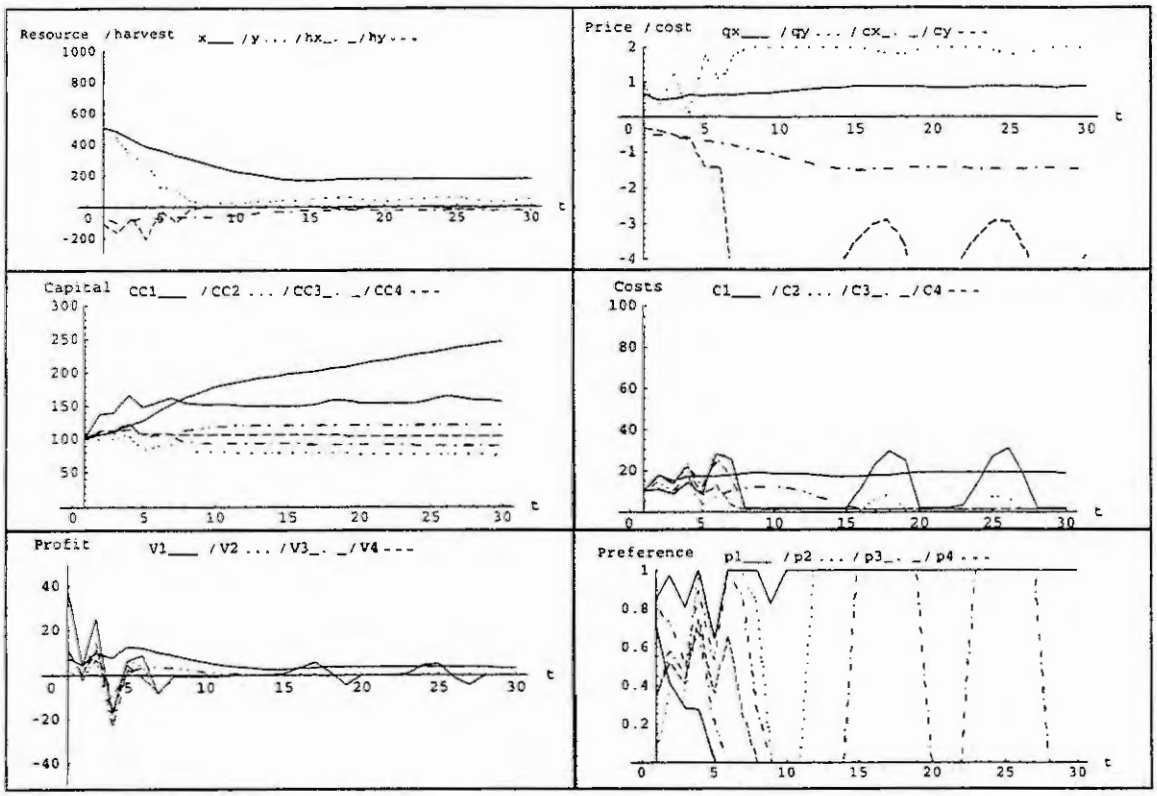
Case 2 (cooperative sustainable targeting): Now the players no longer aim for maximum profit but for keeping sustainable fish quota and distribution of the costs. This strategy of cooperative sustainable targeting guarantees that the fish resources are stabilized at about 200 units for y and 400 units for x, ensuring a permanent high profit for all players and increasing capital for all. After initial fluctuations in preference space all players have chosen a permanent preference for one resource. All players profit from this deal, as well as the fish populations.

Conclusions

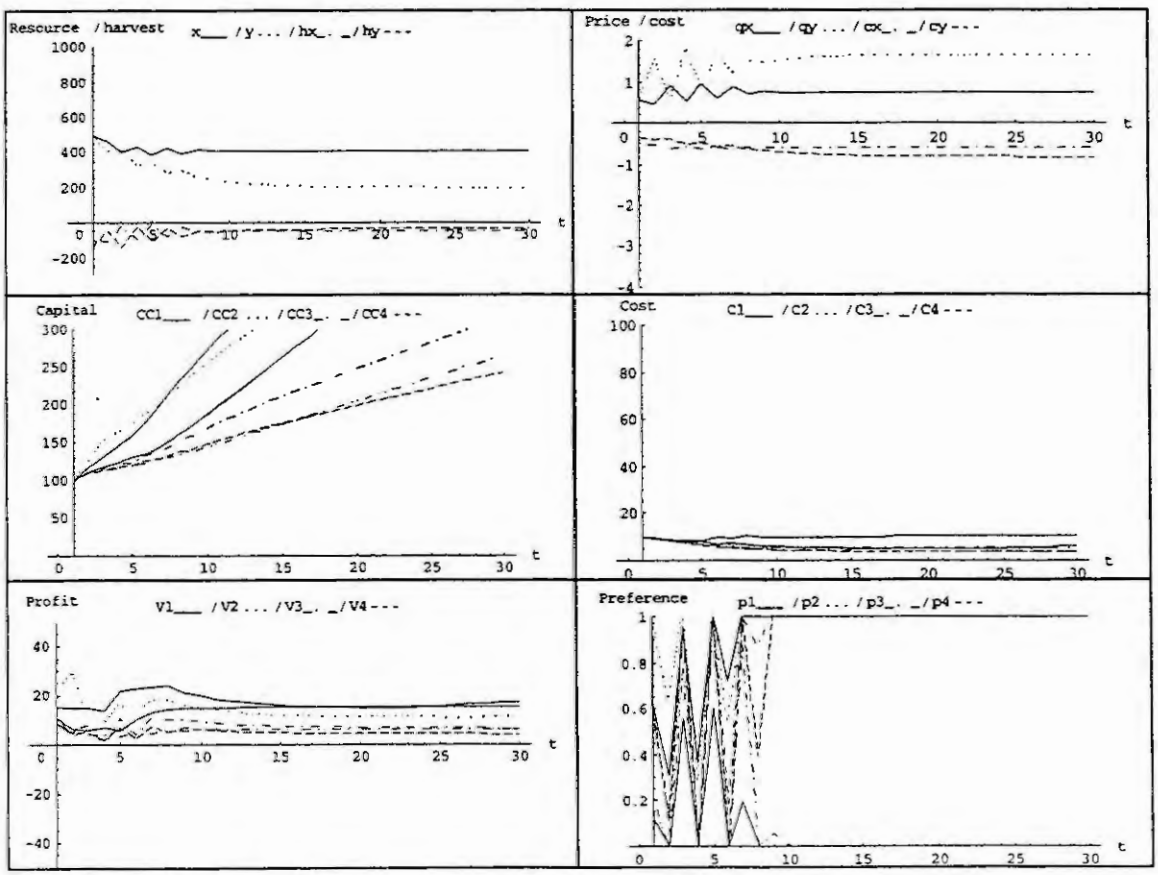
Both the theoretical analysis of resource conflict and the computer simulation of fishery management suggest that highly competitive situations with individual profit maximization provoke a struggle for survival between the resources as well between the players who need the resources for their well-being. Everyone looses, though a few players may have a relative advantage compared to others. Protection of the resource stock by a strategy of cooperative sustainable targeting provides the highest profits as well as capital growth. To achieve this a stronger degree of cooperation and control is required to ensure that sustainable resource quota are not exceeded by individuals. With increasing number of players the possibilities for instability increase and thus the difficulties to control sustainable targeting.

References

1. Carraro, C. and Filar, J.A., Control and Game-Theoretic Models of the Environment (Eds.), Birkhäuser, Boston, 1995.
2. Hanley, N., Shogren, J.F. and White, B., Environmental Economics in Theory and Practice. Oxford University Press, Oxford, 1997.
3. Hofbauer, J., Sigmund, K., Evolutionary Games and Population Dynamics. Cambridge University Press, Cambridge, 1998.
4. Ipsen, D., Rösch, R. and Scheffran, J., Cooperation in Global Climate Policy: Potentialities and Limitations. Submitted to: Energy Policy.
5. Jathe, M., Krabs,W., and Scheffran,J., Control and Game-Theoretical Treatment of a Cost-Security Model for Disarmament. Mathematical Methods in the Applied Sciences, 20(1997), 653-666.
6. Pickl, S., Der tau-value als Kontrollparameter. Shaker, Aachen, 1998.
7. Scheffran, J., Komplexität und Stabilität von Makrosystemen mit Anwendungen. Marburg, 1983.
8. Scheffran, J., Modelling Environmental Conflicts and International Stability. In: Models for Security Policy in the Post-Cold War Era, (Eds.: Huber, R.K. and Avenhaus, R.) Nomos, Baden-Baden, 1996, 201 – 220.
9. Scheffran, J., Environmental Conflict and Sustainable Development. In: Environmental Change and Security, (Eds.: Carius, A. and Lietzmann, K.M.) Springer, New York, 1999, 195 – 218.
10. Scheffran, J., Modelling Sustainable Use of Natural Resources. Proc. SOR99, Magdeburg, 1999.



Case 1: 6 competing value-maximizing players, random preferences, varying efficiencies



Case 2: 6 players with cooperative sustainable targeting, .

Figure 3: Simulation of the fish resource conflict for two fish populations and six players

List of Authors

VIRTUAL REALITY: A METHODOLOGY FOR ADVANCED MODELING AND SIMULATION OF COMPLEX DYNAMIC SYSTEMS

Dietmar P.F. Möller
*University of Hamburg, Department Computer Science, Chair Computer Engineering
& McLeod Institute of Simulation Sciences, German Chapter at University of Hamburg
Vogt-Kölln-Str. 30, D-22527 Hamburg, Germany
Tel.: ++49-40-42883-2438. Fax: ++49-40-42883-2552,
Email: dietmar.moeller@informatik.uni-hamburg.de

Abstract. We present the methodology of virtual reality, as a framework for advanced modeling and simulation of complex dynamic systems. A framework at a glance deals with the general purpose capabilities of the methodology behind. From this point of view, the virtual reality framework could be introduced as a natural domain for collaborative activities because virtual reality allows users doing things they normally can not do in reality. Due to that point our claim is that our work should give insight into complex realized research study examples showing the power of virtual reality methodology in the spacious area of complex dynamic systems. Based on that facts our work will give case study examples of virtual reality applications in nanotechnology as part of an embedded interactive dynamic process.

For this areas, the inclusion of a metaphor incorporating the notation of a virtual world provides significant enrichment. In general, virtual reality as methodology for virtual environments allow sharing simulations.

1. Introduction into Virtual Reality as Methodology for advanced Modeling and Simulation of complex dynamic systems

As in real science research laboratories and/or design centers, as well as in the manifold of industrial research and design centers, people must be able to talk to each other, move around, connect equipment, build up test sets for the respective devices under test, design devices and/or system highlight points of interest for others to consider, and jointly edit documents, reports, and 3D-models. The ultimate promise of virtual reality though is that users will be able doing things they normally can't do in reality. Our motivation for virtual reality and hence virtual environment is to go one step beyond ubiquitous computing.

The basic idea of ubiquitous computing, is that a single computer should not be the locus of computation in one's research laboratory, research and design center, business or other environment [Möller, 1998]. Technology should be embedded and/or distributed in the environment in an invisible as well as in a transparent way. Within this virtual reality environment there would be lots of computationally driven gadgets or so called smart applications throughout, each one could be part of a larger system of combined de-vices, receiving and transmitting signals as from abroad or from intrinsic systemic pathways.

In simulation science the methodology of Virtual Reality, short VR-world, offers possibilities for

- points of collaboration and common interests
- maintaining environmental coherence across room and/or lab changes
- hardware, software and product design
- education and training.

Virtual Reality as methodology for virtual environments, based on simulation, is a natural domain for collaborative activities because virtual reality allow users doing things they normally cannot do in reality, e.g. being within a molecule, being inside the combustion chamber of an automobile engine, walking through a tunnel in „outer space" etc.

Our goal in simulation in virtual reality is to unite the power and flexibility of virtual reality methodology with the insight of ubiquitous computing which can be stated as computation in space and time, based on:

- image processing
- photogrammetry
- computer graphic and visualization
- 3D-modeling based on innovative algorithms
- synthetic scene generation.

Due to intuitive interaction within the virtual reality methodology, normally arousing interface problems are not the point of conflicts. This is of importance in big transdisciplinary projects. Moreover the integration of modern 3D-control mechanisms like head mounted devices, cyber gloves, spaceball etc. enable users navigation

through virtual worlds. The effect of immersion, which means the realization of space depth, allows the user a very fast adaptation to processes in space and time.

Due to that fact, very new scenaric presentations are possible, containing branch specific elements and knowledge, e.g. in geoscience, geotechnology, etc. Hence virtual reality offer a concept for modeling and simulation of complex systems with parametric as well as nonparametric topology within a unique frame. This results in rapid prototyping, based on flexible virtual reality modeling tools with concepts for geometry, motion, control, as well as virtual reality components like images, textures, voice, animation, multimedia, video, etc.

The technical complexity associated with developments in the virtual reality domain require for the introduction of characteristics of metric values. In the development of utility metric values, several important factors that relate to the metric values itself must be considered, especially metric dimensionality, metric attributes, metric types, etc.

A very easy and most straightforward approach for realisation of metric valuated dimensions could be found using unidimensional scaling. Methods of unidimensional scaling, however, are generally applied only in those cases where there is good reason to believe that one dimension is sufficient.

But metric valuated accuracy and presentation fidelity leading for a multidimensional scale. A multidimensional scale is necessary for an adequate image's quality description, if additional information would probably be required. Therefor a multidimensional scale must be developed. Metric valuated attributes are the actual quality parameters measured along each quality dimensions, which are realism, interpretability and accuracy.

There are a number of possible metric valuated types that could be used for the dimensions of a quality assessment metric. Referring [8] to these types are

- criteria based (on a textual scale which define the levels of the scale)
- image based (on a synthetic scene where a rating is assigned by identifying the standard image having a subjective quality that is closest to that being rated)
- physical parametric based (on measured values based on integrated power spectrum, or mensuration error, etc.)

2. Virtual Reality Methodology in the area of Nanotechnology

Our recent virtual reality investigations due to the research area of nanotechnology deal with ultra-fast energy transfer dynamics between semiconductor surfaces and adsorbed chromophores. The technique of near-field scanning optical microscopy, so called NSOM, offers the possibility of high spatial (Äx < 50 nm) and temporal (Ät < 100 fs) resolution while retaining the advantages of optical spectroscopy.

Based on a pump-probe laser equipment we received 3D (x, y, Ät) images as a result of changes in the probe beam transmittance or reflectance which is a function of pump-probe delay (Ät).

NSOM is a scanned probe microscopy system, in which light from a tapered fiber optic located close to a surface, allows illumination on sub-wavelength dimension. Experiments showing femtosecond time-contrast NSOM have not yet been accomplished. Silicon-based devices in common electronics (e.g., the Pentium chip) already rely on features in the 350-nm size range. As the evolution towards smaller, and faster devices continues, measurement techniques must be developed which probe the chemistry and physics of interfaces on shorter spatial and temporal scales simultaneously.

Figure 1a-d shows two examples of a feature extraction which is done by a two-step successive class-oriented average difference autopower spectra calculation. Non important portions of information that would have been part of the neural classifier's decision-basis (shown on the left-hand side) will be eliminated, and the important system information is thrown into relief (shown on the right hand side).

Due to the high temporal and spatial resolution of the pre-processed pattern it is possible to investigate dynamic processes in semiconductor materials in virtual reality, e.g. the energy transfer between adsorbed molecules and their topologic relations, the affect of the binding condition of single molecules and the effects of defects on the semi-conductors surface by matching the object surface with soft-computing based classified classes in a virtual scope domain in detail. While the knowledge of those interacting and non-interacting sub-processes is incomplete, at the very first their correlation is analysed by so called selforganising maps (SOM) as this method of a non-supervised learning strategy in the virtual space of neural nets, which will allow to detect the correlation between contributed sub-systems, even if the knowledge of them is incomplete.

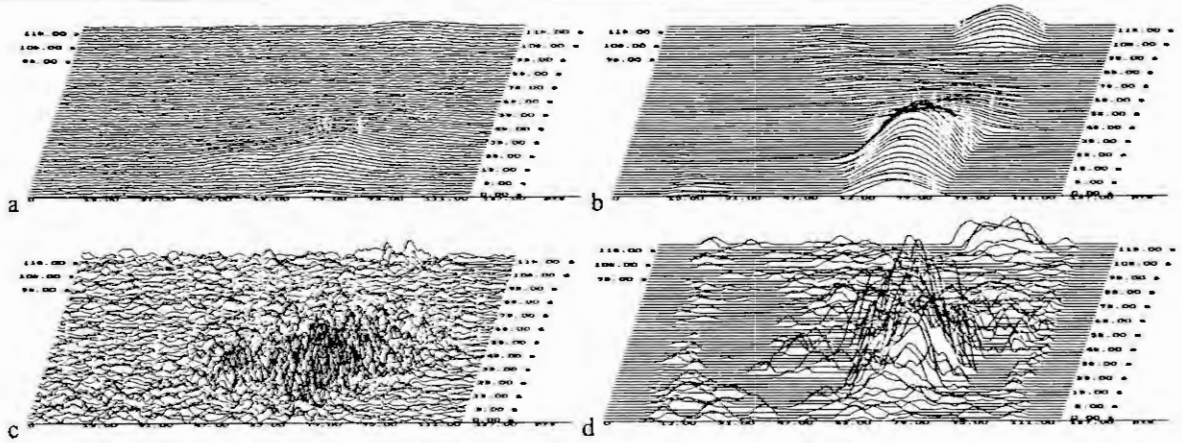


Fig. 1: Spectra of emission on a semiconductor surface:
a: Defect on surface without pre-processing procedure,
b: Defect on surface pre-processed by difference power spectra,
c: Pump-impulse without pre-processing procedure,
d: Pump-impulse pre-processed by difference power spectra

Due to the characteristics of selforganized maps we also expect to detect structures and dependencies of transisent dynamic data which are encoded in input vectors as especially in vast and muddled data sets. Hence SOM are an outstanding tool to verify inner relationships.

As a very first interim result in context to the present problem analysing ultra-fast energy transfer processes between single molecules onto a semiconductor surface, defects on the surface are identified as a permanent background noise contribution that masked the emission spectra of the pump-probe experiment.

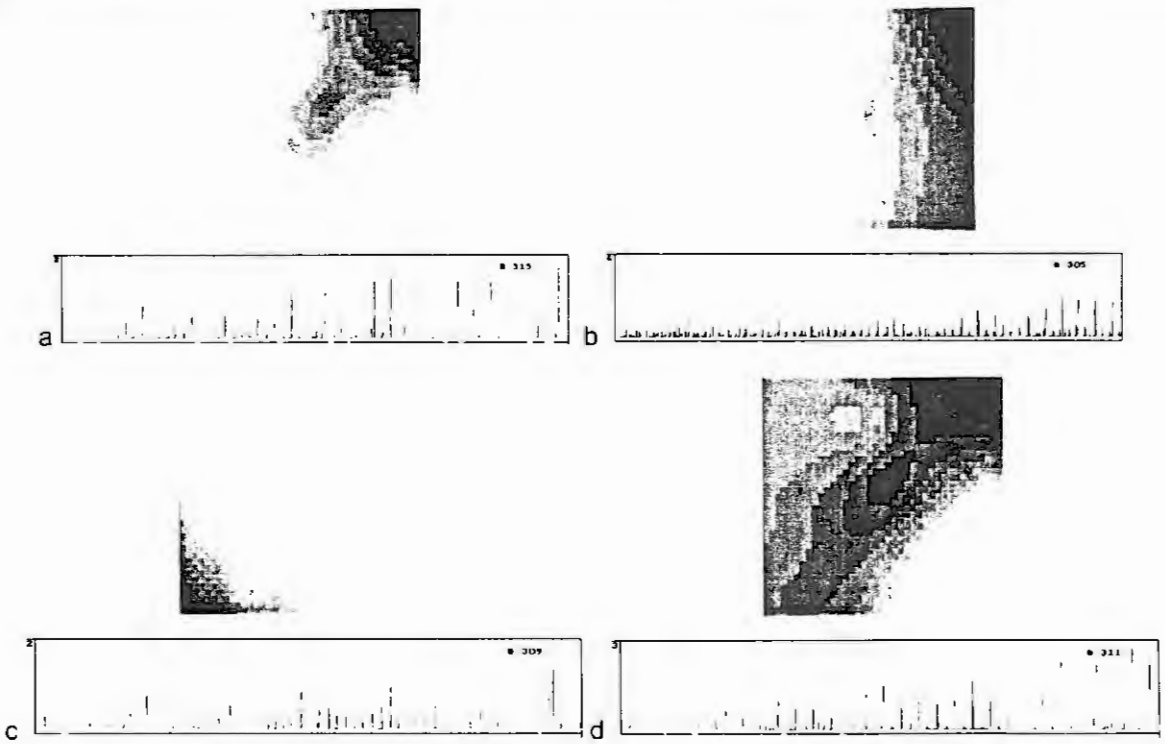


Fig. 2: representation of the NSOM experiment on a self-organising map (SOM)
a: Initial sequence, b: Pump-impulse occurs,
c: Sequence of restoration, d: Final sequence

Figure 2 presents a temporary extract of snapshots of the semiconductor surface as input vectors (spectra below) and their representation as areas of activity (light colored) on the SOM. The topographic patterns are each presented to a SOM (with 40*40 neurons) as single vectors that consist of all successive rows of the emission spectra-scan.

As the initial reps. final sequence of the measured time interval Figures 2a and 2d show similar structures of the areas of activity due to the fact that the emission spectra are restored completely. Figure 2b depicts the emission spectra when the pump impulse occurred. Though a new center of activity has been established on the SOM, the activity-area of the initial sequence has not been destroyed completely and is participate at the actual spectra as an independent system-parameter. The decrease of influence of the pump impulse as a consequence of the increasing time gap between pump and probe impulse is distinctly expressed by the smoothly migration of the center of activity towards its initial structure.

To detect the exact moment when the pump impulse influences the emitted spectra, a backpropagation neural net type was trained with the same pre-processed data, while the optimal choice of the learning vectors are derived by the results of the SOM analysis. In Figure 3a the period of learning of this neural net type is shown. The gap between the two curves marks the degree of probability of identification while the upper border of the gaps represent a probability of '1' and the lower border a probability of '0'. The first surprising result is, that after a period of less than 100 learning steps the classification behavior of the net satisfies and the net was ready for action.

The temporary identification of the pump pulse in an infinite slope done by this net is shown in Figure 3b while the time of its influence is also a matter of interest.

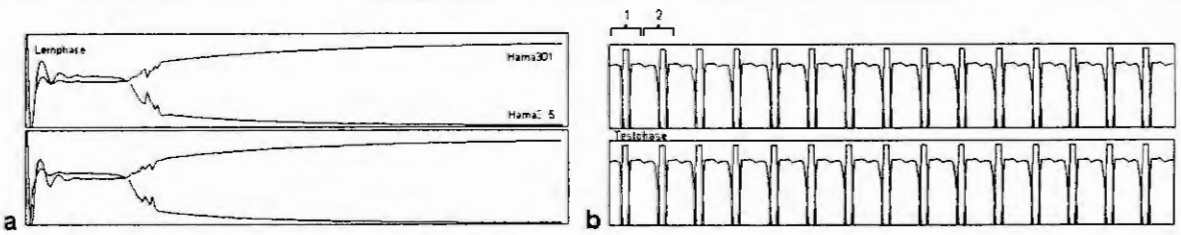


Fig. 3a,b: Classification of the emission spectra by a backpropagation net

As a very first result we can state that the virtual reality inspector neuronal network is able to clearly identify that the pump impulses can be assumed as columns which occur periodical.

3. Conclusion

The potential of virtual reality is huge. We only scratched the surface of the complex space that potentially contains an incredible number of solutions to the problem of how best to design systems/devices/processes etc..

4. References

1. Crilly, A.J., Earnshaw, R.A. and Jones, H., Applications of Fractals and Chaos. Springer Verlag, Berlin, 1993.
2. Encarnacao, J.L., Peitgen, H.-O., Saka, G. and Englert, G., Fractal Geometry and Computer Graphics, Springer Verlag, Berlin, 1991.
3. Gilfillan, L. and Harbison, K., Using distributed virtual environments (DVE) for collaborative program planning and management: Problems and potential. In: Proc. VWSIM'98 (Eds.: Landauer, C. and Bellman, K.L.), SCS Publishers, San Diego, 1998, 39-46
4. Möller, D.P.F., Virtual Reality: Simulation Synergy in Laboratories and Outer Space Domains. In: Simualtion: Past, Present and Future (Eds.: Zobel, R. and Möller, D.P.F.), Vol. II, SCS Publishers, Delft, 1998, 64-66
5. Schneider, M., Spatial Data Types for Database Systems. Springer Verlag, Berlin, 1997.
6. Singh, A., Goldgof, D. and Terzopoulos, D., Deformable Models in Medical Image Analysis, IEEE Press, Los Alamitos, USA, 1998.
7. Straßer, W. and Seidel, H.-P., Theory and Practice of Geometric Modeling, Springer Verlag, Berlin, 1989.
8. Yachik, T.R., Synthetic Scene Quality Assessment Metrics Development Considerations. In: Proc. VWSIM'98 (Eds.: Landauer, C. and Bellman, K.L.), SCS Publishers, San Diego, 1998, 47-57

VIRTUAL REALITY VISUALISATION: A NEW METHODOLOGY FOR MINIMAL INVASIVE CARDIAC SURGERY

E. Godehardt[1], D.P.F. Möller[2], B. Kesper[2], P.Feindt[1], J.-A. Koch[1]
[1]University of Düsseldorf,
Moorenstr. 5, D-40225 Düsseldorf
godehard@uni-duesseldorf.de
[2]University of Hamburg
Department Computer Science
Vogt-Kölln-Str. 30, D-22527 Hamburg, Germany
{dietmar.moeller, kesper}@informatik.uni-hamburg.de

Abstract. The treatment of patients with myocardial pumping insufficiency results, under normal circumstances, in heart transplantation. We focus on the possibilities of morphing in order to model patients' hearts, based on measurements obtained from the CT and MR. The derived morphing model will be the basis of a two-fiber elastic network design, which should be pulled tight over the heart of a patient. Based on this cardioplasty, the contractility of the ventricles will be more sufficient, and transplantation may be avoided.

1 Introduction

The treatment of patients with myocardial pumping insufficiency results, under normal circumstances, in a maximal cardio surgery case, the heart transplantation. Heart transplantation covers a high risk for the patient, and a longlife postoperative specific lifestyle, as well as huge costs for the surgical event. Due to this situation, we focus on the possibilities of morphing, a specific virtual reality methodology, for 3D volumetric modeling, in order to morph the human heart intra-individual, based on measurement obtained from the CT (computertomography) and MR (magnetic resonance). The derived morphing model will be the basis of a two fibre elastic network design, which should pull tight over the heart of the patient, inserted during minimal cardio surgery. Based on this cardioplastie, the contractility of the ventricles will be more sufficient.

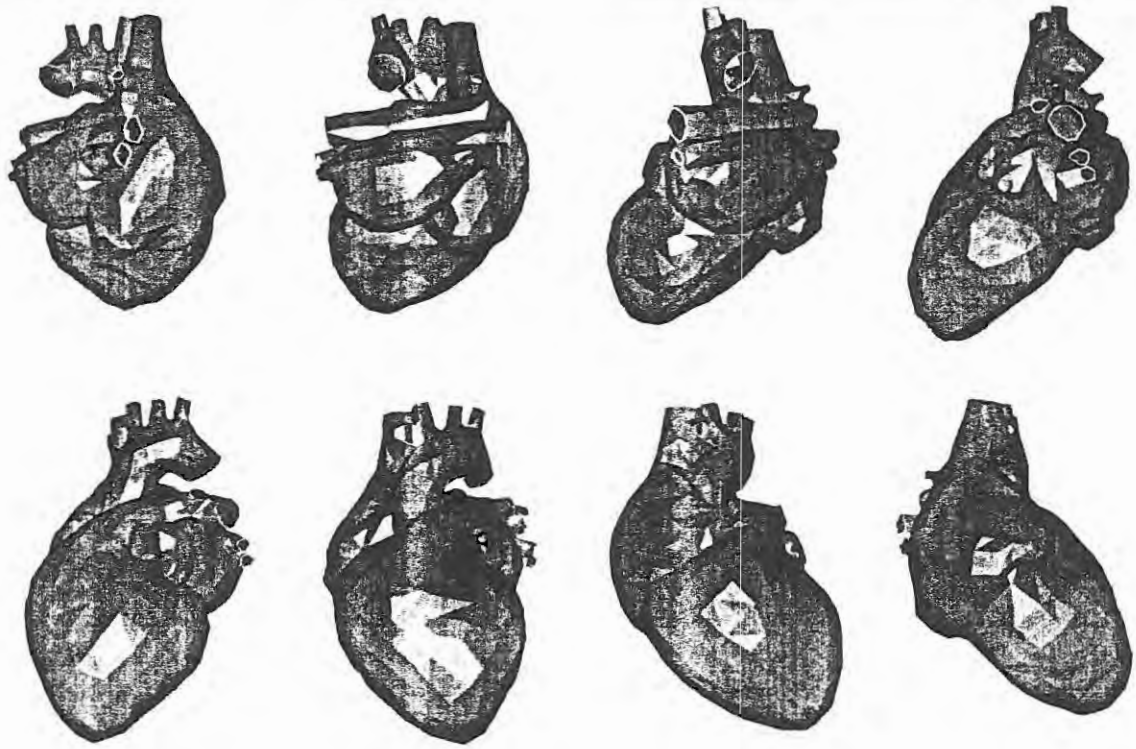


Fig. 1: Rotating human haert

2 Segmentation of Inner Organs

CT and MR measurements result in very good representations of inner sections of the human body. Nevertheless the segmentation of separate organs, and the representation of real 3D reconstructions based on soft shapes from the measurements, is up to now, a task, which had not been solved in a sufficient manner.

The different segmentation methods are based on the assumption, that the grey values of the measured dot space can be interpreted as the border between organs. Conventional methods are based on dot pursuit algorithms which have the important disadvantages of:

1. The resulting models are primarily voxel based, and due to that fact limited for carry on the signalprocessing procedure
2. If grey values are within gaps, they will be considered within the 3D model, but an actual „polish fit" is necessary
3. Vague data of in vivo as well as in vitro measurements of organs, like heart, liver, lung, etc. result in a significant deterioration, and hence the extracted 3D model will be inaccurate.

Treating the disadvantage of the methods discussed above, we developed, for the area shape reconstruction of the human heart, a specific morphing algorithm, the so called Multi Level B-Spline Approximation for 3D modeling. The data, obtained from the respective CT and MR measurements are being weighted in accordance to the grey values which, thereafter, being treated as a projection onto a free space allocation area, the Non Uniform Rational B-Spline, the so called NURBS. Due to the mathematical representation within the projection reference point in the direction of the vector of the projection, the given space domain will be deformed in a successive manner one after the other. By that, the influence on the dot projection will be more severe weighted and thereby global more effective onto the respective free space allocation area. In a sequence of steps, the influence of weighting factors will be more and more reduced, hence the deformation will be effect only local areas of the representations in between.

Changing the order of the polynoms of the original space description function, and the number of iteration steps, used, allows a problem dependent approximation of the dot space dependent information description.

Using appropriate weighting factors, and simultaneous distortion of the models through all the dots, single and remote dots will be smoothed automatically, whereby also from the CT and MR pictures 3D models can be extracted. Moreover, as an initialisation, a rough approximation of the human normal heart can be extracted, and hence, the clinical expert knowledge will influence the process of reconstruction. Due to that intrinsic power of the methodology, used, after a few iteration cycles, an exact description of the heart is possible.

Based on constraints of the most important dots, single sub domains of the outcome space domain can be manipulated. Based on the knowledge of the position of single organ compartments, e.g. the ventricle of the right heart, the target oriented deformation of the basis space domain is possible.

The methodology, developed, fits as well in the situation of less data, as well as in the situation of huge data sets. In case of missing data sets as well as vague data sets, the methodology, developed, is able to process the data sets in an automatic manner, due to the fact, that locations with ill defined data sets, need no more deformability, after a couple of iteration steps.

Modeling with NURBS free space domains allow simple and easy correction of the model by the physician, as user. Simply pulling the generated space domain, and justification of the dependencies within the influence domain, the model can be much more completed or much more detailed compartments of the heart can be modeled in a manual manner.

For a better understanding, the generated model can be used with the embedded virtual reality visualization environment. Multimedia aspects like figures, animation, etc. are also easy as well as powerful to use.

3 Evaluation

The morphing methodology, described above, had been used in animal research studies with 12 pigs, body weight in between 40 and 45 kg.

After an atrial pacing to release a left ventricular pumping heart insufficiency, a static-dynamic fibre-elastic network, which was, pull tight over the left ventruculus, located, as a strong indication for the therapy of dilatation of cardio-myopathia. Based on the virtual reality 3D representation, the fibre-elastic network can be intra-individual adapted. At the very late stage of the experiments, the heart will be extracted and the real data sets measured, for re-evaluation.

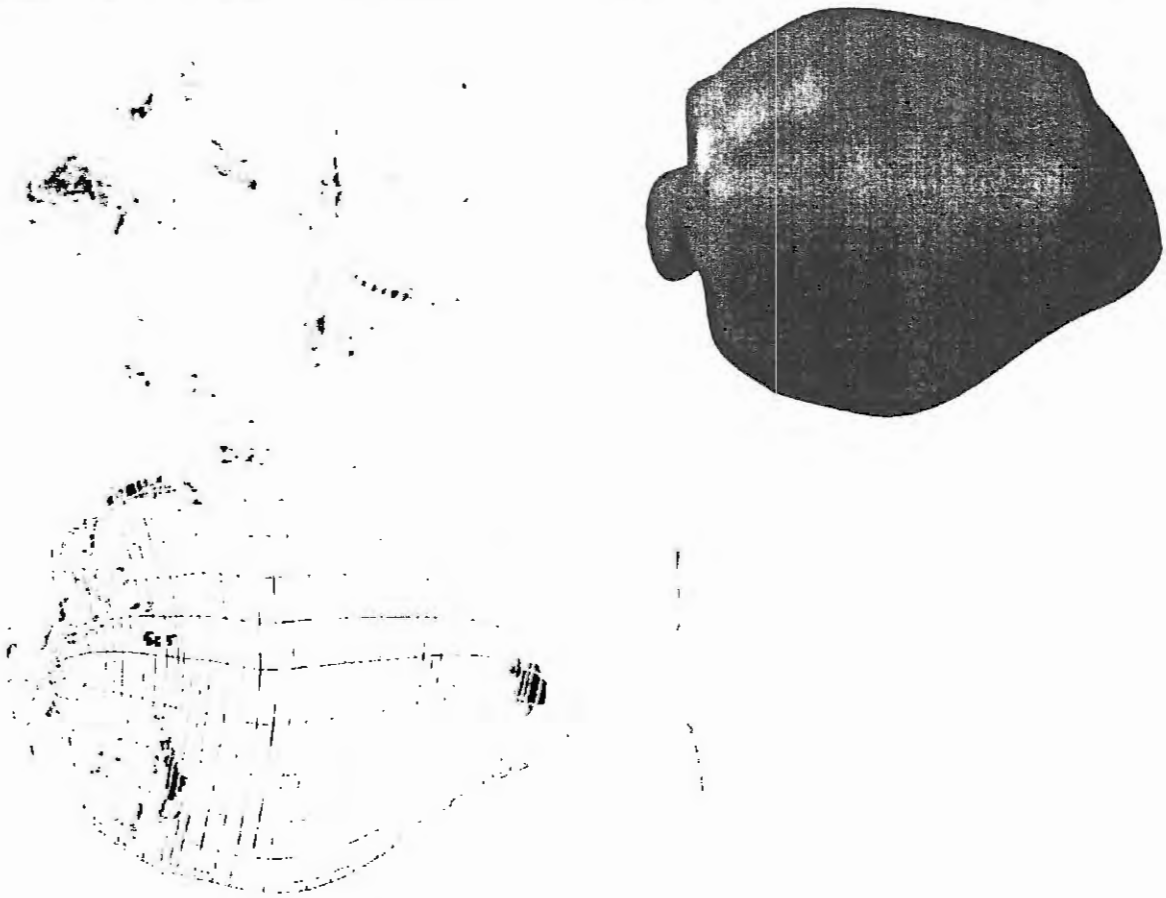


Fig. 2: The morphing approach applied to a pig heart and the resulting tailord net

4 Clinical Approach

After the successive animal experiments, the use within the university hospital is planned. Preoperative a single exact measure of the myocardial dimension for the elastic fibre network is necessary, based on MR shoots of the heart surface of intra-individual patients, as well as the measure of the geometry, based on the new developed morhing methodology will be done. Moreover the haemodynamic changes, based on the net's effects, will be predicted based on a compartment model, and simulated within the virtual reality tool box system.

In case that the virtual reality simulation studies results in a sufficient influence of the haemodynamics of the respective patient, based on the data obtained, a stereo-lithographic model of the heart will be designed and manufactured. This network, pull tight to the heart muscle, will be manufactured based on two different elastic fibres, due to fit as best with the individual myocardial plastic. The first fibre is stiff and correlated with the maximum of expansion of the heart. The second fibre corresponds against the maximum of expansion of the heart; du to this situation the heart muscle will be assisted during the pumping process which start in the very early beginning of the contraction cycle.

We expect with this technology a longterm reduction of the pressure on the hearts muscle, which is of importance due to the lack of time schedule possibilities for heart transplantation, as well as a continuous recovery of the heart. In this case the heart transplantation will not be necessary, due to the very well assist of the elastic fibre network, which results in a drastic reduction of costs, and in a huge number of patients, which can be supported with this new technology, as well as patient with an contra-indication of a heart transplantation. The static cardio-plastie can also be used to prevent ventricular aneurysmia after a myocardial infarct, as well as in the very early stage of cardio-myopathia, in order helping the disabled.

5 References

1. Albers, J., Schroeder, A., Makabe, M.H., Gaa, J., de Simone, R., Vahl, C.F. and Hagl, S., Validierung digitaler Volumetrie und dreidimensionale Rekonstruktion kardialer magnetresonanz- und computertomographischer Bilddaten: Studie an explantierten Schweineherzen. Kardiovaskuläre Medizin 2, 1998, 96.
2. Frazier, O.H., New Technologies in the Treatment of Severe Cardiac Failure: The Texas Heart Institute Experience. Annals of Thoracic Surgery 59, 1995, 31.
3. Frazier, O.H., Macris, M.P. and Radovancevic, B. Support and Replacement of the Failing Heart. Lippincott-Raven Publishers Philadelphia – New York, 1996.
4. Lee, S., Wolberg, G. and Shin, S.Y., Scattered Data Interpolation with Multilevel B-Splines. IEEE Trans. On Visualization and Computer Graphics, Vol. 3, No. 3, July–Sept. 1997.
5. Singh, A., Goldgof, D. and Terzopoulos, D., Deformable Models in Medical Image Analysis. IEEE Press, Los Alamitos, USA, 1998
6. Wabel, P. and Leonhardt, S., A Simulink Model for the Human Circulatory System. Biomedizinische Technik 43, 1998, 314

VIRTUAL REALITY MODELS FOR ADVANCED SIMULATION IN GEOSCIENCE AND GEOTECHNOLOGY

B. Kesper[1], D.P.F. Möller[1], G. Reik[2], C. Zemke[2]
[1]University of Hamburg
Department Computer Science
Vogt-Kölln-Str. 30, D-22527 Hamburg, Germany
{kesper, dietmar.moeller}@informatik.uni-hamburg.de
[2]Technical University of Clausthal
Institute of Geology and Paleontology
Leibnizstraße 10, D-38678 Clausthal-Zellerfeld
{zemke, inggeo}@geologie.tu-clausthal.de

Abstract. Virtual reality as a tool for interactive three- and four-dimensional visualization of natural or technical domains has increasingly developed over the last few years. On the background of geoscientific and geotechnological application domains we present the framework of an integrated information system based on extended database concepts, an embedded spatial model and integrated temporal aspects that is brought together in an advanced virtual reality system for the simulation, prognosis and analysis of *space in time*.

1. Introduction

Having its roots in computer graphics and simulation, the methodology of virtual reality is applicable in many different problem domains, ranging from industrial and military applications to academic research [6]. Based on features offered by computer graphics to visualize highly realistic models [2] and through the integration of real time computing virtual reality enables the user to move around in virtual application spaces. By embedding temporal concepts virtual reality can be used as a basis for simulation, analysis and prognosis of complex processes. Furthermore underlying databases offer the ability to efficiently store and retrieve huge amounts of data for the modeling of real world domains. Thus, virtual reality can be seen as an embedded system that combines different approaches in one integrated solutions.

Nevertheless state of the art virtual reality technology misses the capabilities of real simulations of complex models with the three aspects

- thematical characterization,
- space and
- time.

Although the need to combine these three aspects and to simulate with regard to this information, only few approaches to solve this problem have been developed [5]. All these developments miss the capabilities of real dynamic simulations that allow the user real time interaction not only within the three-dimensional model itself, but also within the parameterized processes, thus leading to a better framework for complex system analysis.

This paper gives an insight view to a complex research study example showing the power of virtual reality as a basis for advanced simulation of space in time.

2. Geoscience Applications

The focus of geoscientific and geotechnological research in geology, engineering geology or hydrology nowadays concentrates on computer simulation and information systems for underground studies, i.e. in soil modelling [1]. Expensive in-situ testing has been replaced by accurate computer simulations. This not only reduces the costs, but simulation furthermore can calculate results within minutes and give first hints, why certain effects occur. These hints than can be used to achieve even better simulations and can help to better understand the complex geo-dynamic processes.

Geoscientific research with the aid of computer science up to date thus can be parted into two major categories:

1. *Spatial Information Systems* with the main purpose to efficiently store, analyze and display thematical and geometrical data. Especially geologists and geo-engineers use spatial information systems to create three-dimensional models of technical entities and the surrounding geological underground which help them to explore the environment. As the word information system already implies, these systems in most cases only help to get a better information-retrieval based on the information stored in a database. The data can be visualized in rendered 3D-models, but real user interaction is not provided.

2. *Process Simulation*, like fluid flow or chemical reaction analysis, is concerned mainly with solving some kind of differential equations. These are specialized for more or less one purpose, so that the input data has to be extracted from real world measurements or information systems. The results are presented in tabular listings, charts or very seldom in spatial models.

Our goal is to combine these two approaches with the background of the virtual reality methodology resulting in one unified system for dynamical virtual reality simulations of space in time (Fig. 1).

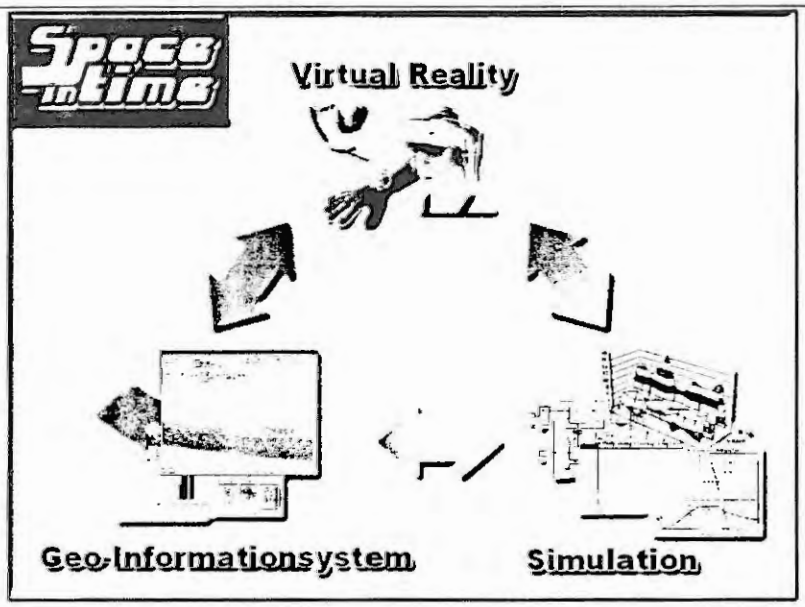


Fig. 1: Unified System Approach

3. Modeling and Simulation of Space in Time

The process of modeling in geo-applications differs from computer aided design (CAD) where the user develops a new car based on idees and given restrictions. In contrast to this construction process, creating a three-dimensional model of a geological underground is dealing with re-construction of a complex part of the real world that furthermore is only known in small pieces, i.e. through borehole drilling etc. The geoscientist tries to re-model a part of the real world domain as accurate and realistic as possible.

The next step is to integrate temporal aspects representing the processes and the development of the geological underground that finally leads to real dynamical virtual reality models. To accomplish this task temporal database concepts for the management of time and different scenarios were developed with regard to the geoscientific background. Temporal aspects, in this approach, can be seen as

- continuous time by means of a single vectored parameter, and
- time-dependent versioning allowing the simulation and adaptation of three-dimensional spatial models based on various parameters.

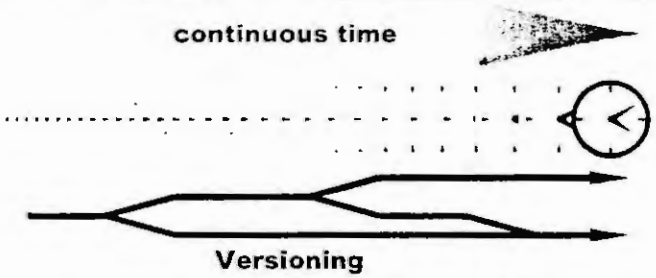


Fig. 2: Continuous time and versioning

These temporal concepts are embedded within the underlying virtual reality database model, allowing the user to build four-dimensional models as a combination of three-dimensional structures and the processes within these structures.

With this integration of temporal information the basis for simulating processes and thus to analyze and prognosticate future developments under differing condition within various scenarios is given. The model itself changes dynamical while the user analyses and interprets the simulation results and changes the input parameters [7].

Within these complex four-dimensional models of space in time, consistency is of major concern. While the user tries to describe a part of the real world, the system has to check weather user-defines constraints are met or if all solids are placed non-overlapping not only in the static three-dimensional model, but furthermore in its changing through temporal development.

In addition, the best results of simulations or analyses are worth nothing, if their calculated values can not be interpreted and remapped to the original real world problem. For the evaluation of complex simulations it is necessary to recheck the results with real world measurements to ensure their correctness. Therefore examples have to be found that give the possibility to determine error values, what in fact can be a difficult task.

Last but not least any simulation result is inapplicable if the data presentation can't provide the new information to the user. Very often the outcome of a complex simulation is presented in numeric tables that cover the results rather than point them out. First approaches for user friendly data visualization use charts and diagrams. The problem arising is the loss of the three dimensional background. The best way to present the results would be the integration within the abstract spatial model of the real world domain. The user then could identify different parameters and their distribution directly at their actual position in space and time.

Based on these assumptions its easy to understand that applications for the geoscientific domain need a higher degree of automation for system-supported model generation. In the near past, this has been neglected in geo-informationsystem development.

Furthermore the use of special simulations and analyses should be kept in mind right from the start. Any data structure for an application domain that allows the input of all needed information but misses the ability to connect, retrieve and compute based on the stored information with respect to clearly defined goals can not achieve efficient data management.

4. The virtual reality core

A first system approach for spatial information management has shown that movement in an artificial three-dimensional model is quiet difficult, especially for inexperienced users. Through the integration of virtual reality concepts, the gap between a four-dimensional model and the usually two-dimensional user-interface can be closed [3,4].

We define Virtual Reality as an embedded system of hardware and software components which gives the user the opportunity to view and interact with a virtual model in space and time with the purpose to analyze different scenarios while changing the underlying simulation parameters.

In our view hardware components consist of a computer, 3D-input and –output devices such as Head Mounted Displays (HMDs), Cyber-Gloves, Head-Tracking systems, and eventually some sort of measuring equipment for real-world data input.

Software in our context consists of tools for 3D-modelling, realistic rendering, imaging, photogrametrie, simulation and analyzing as well as an embedded database management system for the storage and retrieval of huge amount of spatial, temporal and thematical data.

The architecture of the virtual reality core given in Fig. 3 shows the partitioning into three integrated modular layers.

On the *internal level* the basic database management is accomplished. This includes essential concepts, such as transaction management, concurrency control, recovery mechanisms and data retrieval.

The *external level* is a highly integrated user interface that combines aspects of well known user interaction through dialogs in conjunction with advanced virtual reality environments. Using VR-devices such as head mounted displays, cyber gloves and 3D mice (SpaceBall, SpaceMouse), the user can navigate through space and time of the artificial model of the geoscientific real world domain, retrieve information at any point in the model and directly interact, i.e. change parameters or the model itself.

The embedded modeling and simulation is realized on the *conceptual level*. Within this level temporal, spatial and thematical information is represented with the aid of the proposed concepts. The object-relational approach allows any represented part of the real world domain to dynamical change through space and time. As these operations are achieved with methods, the objects themselves behave on the basis of their internal status, the overall model and the interaction of the user.

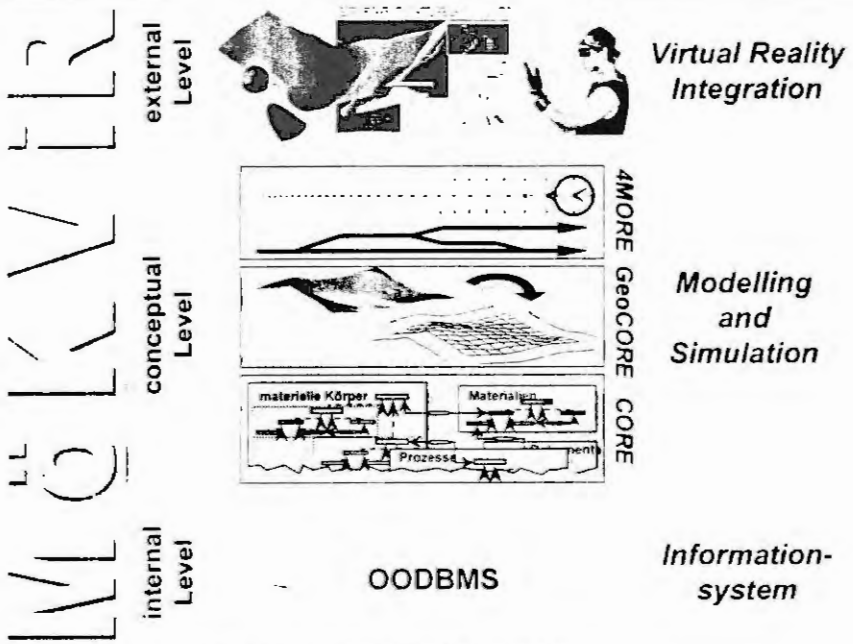


Fig. 3: The virtual reality core architecture

5. Conclusion

We have introduced the concepts for dynamical virtual reality simulation of space in time in the geoscientific application domain. The resulting advanced virtual reality system is a first approach to achieve real user interaction in non-static environments by integrating three-dimensional models and concepts of time as a basis for simulation and analysis of complex geological and geotechnological structures.

The next steps will be the integration of tactile feedback into the virtual reality core, resulting in a system for augmented reality that can be used for research, education and tele-presence applications. With these aspects put into practice geoscientific simulation will have a higher impact based on the increasing interpretability of results through virtual reality models.

6. References

1. Ameskamp, M., Three-Dimensional Rule-Based Continuous Soil Modelling, University of Kiel, Paper 9701, 1997
2. Foley, J., Computer graphics: principles and practice. Addison-Wesley, 1996.
3. Kesper, B., Möller, D.P.F., Reik, G. and Zemke, C., Dynamical Virtual Reality Models for Geoscience Simulation. In: Simulation in Industry, 11th European Simulation Symposium, Erlangen, SCS, 1999, 345 - 348.
4. Möller, D.P.F. and Kesper, B., Virtual Reality: A Methodology for Advanced Simulation in Geoscience. In: Proceedings SCSC 99, (Eds.: M-S.Obaidat, A.Nisanci, B.Sadoun), SCS Publication, San Diego, 1999, 645-649.
5. Schneider, M., Spatial Data Types for Database Systems, Springer Verlag, Berlin, 1997
6. Vince, J., Virtual Reality Systems, ACM SIGGRAPH Books Series, Addison-Wesley, 1995
7. Zemke, C., Reik, G., Kesper, B. and Möller, D.P.F., Parameterderivation for the Simulation of hydrodynamic Processes in Joint Aquifer based on VR Methodology. In: Simulation in Industry, 11th European Simulation Symposium, Erlangen, SCS, 1999, 341 - 344.

Virtual Reality in Modelling and Simulation of Hydrodynamik Processes of Dams

Christian Zemke[1)], Gerhard Reik[1)], Björn Kesper[2)], Dietmar P.F.Möller[2)]

[1)]Technische Universität Clausthal, IGP, Abt. Ingenieurgeologie
Leibnizstraße 10, D-38678 Clausthal-Zellerfeld
{inggeo, zemke}@geologie.tu-clausthal.de

[2)]Universität Hamburg, Fachbereich Informatik, AB Technische Informatiksysteme
Vogt-Kölln-Str.30, D-22527 Hamburg
{dietmar.moeller, kesper}@informatik.uni-hamburg.de

Abstract. In the research field of Engineers Geological Science the use of computer aided information- and management systems that includes modern visualisation and database technologies gets more and more important. The simulation and controlling of anthropogen influenced processes in geo-systems depends on various parameters, which often can only be described by experts knowledge.

The necessity of a reliable judgement of the subsurface of dam sites results of the high potential of damage and also of criteria of economic and ecological relevance. The visualisation and the simulation of processes in the joint aquifer should help to determine the mean systems parameters, and their configuration and weighting in the alternating systems of influence

The hydrodynamic system

The hydrodynamic in fractured rocks depends on combinations of many parameters. Commonly the active system of water bearing element is reduced to the joint planes, because the matrix diffusion becomes irrelevant in nearly impervious rocks. The system fluid ⇔ rock is described by the coefficient of permeability k_f, which has the dimension of a velocity [m/s]. In contrast to the permeability tensor K, which describes only a petrographic character, the features of the fluid as viscosity η and density p are considered too.

$$k_f = K * \frac{p * g}{\eta}$$

with g as the value for acceleration of gravity. (Please note that not the Anglo-Saxon notation is used)

The hydraulic conductance of a single fracture element with a gap of $2ai$ correspond to

$$k_f = (2a_i)^2 * \frac{p * g}{12 * \eta}$$

This relation follows the axiom, that the joint water shows a parabolic velocity profile. In a stationary system the average flow rate in a single fracture is then

$$v_f = k_f * i$$

with i as the value for the hydraulic gradient along the fracture plane.

Corresponding to law of DARCY the percolation Q in relation to a surface F is given by

$$v_f * F = Q$$

The according percolation in relation to the standard aperture of the fracture follows

$$Q = (2a_i)^3 * \frac{p * g}{12 * \eta} * i$$

which is known as the *cubic law*. Actually the percolation is proportional to the third potency of the fracture aperture.

While the amount of n fracture planes with an average distance d is integrated in the calculation of the flow rate in a tested bore-hole section, the following equation results.

$$\sum_n Q = n * (2a_i)^3 * \frac{p * g}{12 * \eta} * i$$

Concerning the filtration rate by DARCY the coefficient of permeability is given by the relation

$$k_f = \frac{v_f}{i} = \frac{(2a_i)^2}{d} * \frac{p * g}{12 * \eta}$$

The applicability of such an idealised Model conception with a constant fracture gap, hydraulic smooth fracture planes in an infinite expanded one-dimensional system is very reduced in practise. The heterogeneous character or the flow rate and its direction is determined by the varying facing, expanding, aperture, morphology, filling and configurations of the joint network. So it seems impossible, or at least not practicable to develop a global deterministic description of the hydrodynamic processes in joint water systems. Concerning the investigations of the subsurface of dam sites the usage of hydraulic pressure tests (WD-tests) and grout injection tests gained acceptance to characterise the hydrodynamic processes by their symptoms.

Analysing methods

Grouting (or injection) is a procedure to improve the strength properties of the subsoil or to diminish its permeability by the installation of grout curtains (mostly cement suspensions) under dams. The WD test determines the water capacity and the flow rates in fractured rocks by varying pressure regimes.

Out of these investigations it is possible to derive different attributes of the joint network, e.g. effective joint apertures. First water capacity of the fractured rock is defined by

$$W = \frac{Q}{c * p_0} \qquad [\text{l/min}^{\text{m}}\text{pa}]$$

with c [m] as the value for the length, and P_O for the pressure of the tested bore-hole interval.

According to the axiom of generalised boundary conditions the following assumptions are declared:

- While the influence of the relative roughness in relation to the aperture of fractures is of minor efficiency, even under higher hydraulic gradients the water flow is laminar.
- The analysis is related to a homogeneous and isotropic substitution.
- The groundwater is incompressible, and no erosion processes take place during the experiment.
- The tested bore-hole section is regarded as a line source.

Based on these assumptions a solution of the stationary flow regime is given by

$$k_f = \frac{Q_{WD}}{2 * \pi * L * (h_o - h)} * \ln \frac{r}{r_o}$$

with:	Q_{WD}	Water discharge	[m^3/s]
	L	Length of bore-hole section	[m]
	h, h_O	Pressure head	[m]
	r	Radius of efficiency	[m]
	r_O	Hole caliber	[m]

Out of these investigation no hydrodynamic value that is related to a direction can be derived. Therefore the facing of the joint network elements has to integrated in the analysis of the hydrology.

The percolation of distinct joint network elements of different generation and facing can be determined by special hydraulic pressure tests with accurate well data that are according to the azimute and angle of the tested joint plane. The determination of the average coefficient of permeability of a +/- parallel troop or a single joint plane which cut the bore hole nearly perpendicular is given by the equation

$$k_f = \frac{Q}{2 * L * (\frac{p_O}{\gamma_w}) * \pi} * \ln \frac{R}{r_O} \qquad [m/s]$$

with:	Q	percolation	[m^3/s]
	L	Length of bore-hole section	[m]
	P_O	Pressure	[kN/m^2]
	R	Estimated radius of efficiency	[m]
	r_O	Hole caliber	[m]
	γ_W	Unit weight of water	[kN/m^3]

Now that is die numeric inventory to analyse the different elements of the joint network separated from each other. While these special investigation are only made at some few locations, the results must be transferred on the structural increments of the whole subsurface.

The parameter model

To represent this parameter model a multi-compartement strategy will be defined. Therefore the test results of the single fracture investigation are combined to complex fracture element combinations which describes the effects of that combinations on the hydrodynamics (fig.2).

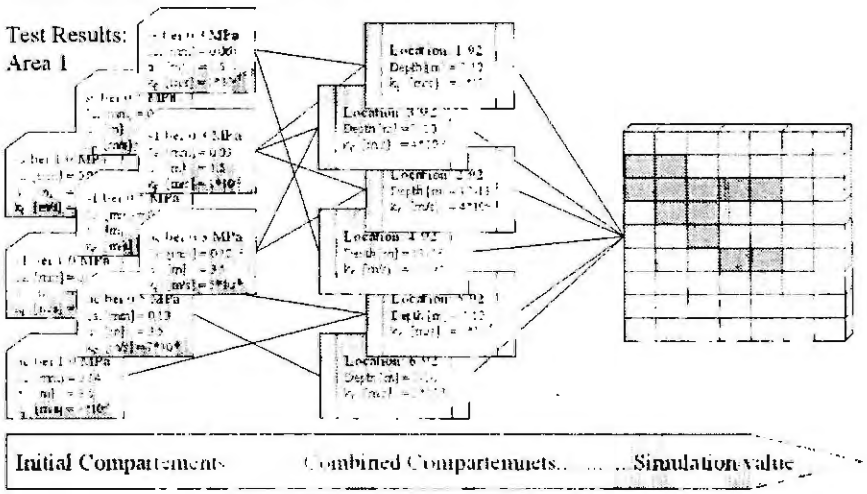


Fig. 2: Basic Multi Compartement Modell

The analyse of the whole system of the subsurface is reserved to the judgement of the expert, what leads to special requests of the man-machine communication. Actually even heavy water bearing elements are of minor

efficiency for the strength properties or effective permeability of the subsurface when their facing is not regarded as critical. On the other hand hydrodynamic processes that endanger the strength properties in worst case, but at least lead to undesired water losses, have to be determined exactly and avoided e.g. by grout injections. To get a better understanding of theses processes, and to make decisions comprehensible, modern techniques of visualisation and simulation have to be integrated in an analysing information system.

Visualisation, simulation and virtual reality

In the actual investigations of the IGP (TU Clausthal) and AB TIS (University of Hamburg) the parameter model of the hydrodynamic interactions in the subsurface of dam sites will be intergrated in the information system BAGIS (fig. 3). BAGIS (Baugeologisch-Geotechnisches Informationssystem) is a product of an interdisciplinary project of the Institute for Geology and Paleontology, the Institute of Surveying and the Institute of Computer Science at the TU Clausthal, promoted by the German Volkswagenstiftung. The general aim of BAGIS is to offer a computer aided information- and management system that includes modern data-base technologies and 3 dimensional modelling and visualisation strategies for the research field of Engineers Geological Science. Hereby the use of NURBS (Non Uniform Rational B-Splines) gives a flexible mathematical description of complex spatial structures and their dynamic.

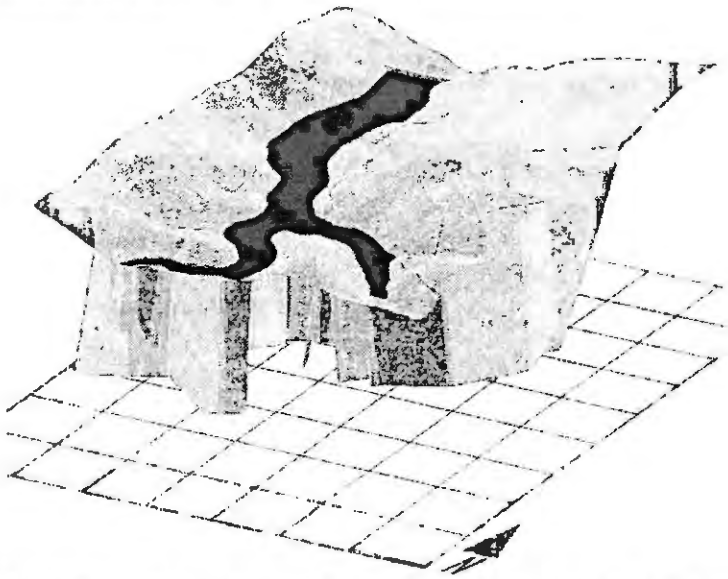


Fig. 3: Initial terrain modell with subsurface fractures and the virtual "Lake Leibis"

The new methods of parameter derivation of hydrodynamic interactions will be integrated in BAGIS as a modern analysing tool on the basis of simulation of dynamic scenario. Concepts of virtual reality will be used as modern interface techniques that suites to human perception.

References

MÖLLER, D.P.F. (1998B): *Virtuelle Realität: Möglichkeiten und Simulation im Laborbereich und Experimental Design*, S. 347-357. In: Simualtionstechnik. Hrsg.: M. Engeli, V. Hrdliczka, vdf-Verlag Zürich1, 1998

MÖLLER, D.P.F.; KESPER, B. (1999A): *Virtual Reality: A Methodology for Modelling and Simulation of Combined Dynamical Systems.* In: Proceedings 4th UKSIM, Cambridge, 7.-9. April 1999, pp. 166-170. Ed.: D. Al-Dabass, R. Cheng. UKSS Publ., Nottingham, 1999.

MÖLLER, D.P.F.; KESPER, B. (1999B): *Virtual Reality: A Methodology for Advanced Simulation in Geoscience.* In: Proceedings SCSC 99, pp. 645-649. Eds.: M-S.Obaidat, A.Nisanci, B.Sadoun. SCS Publication, San Diego, 1999

REIK, G.; ZEMKE, C.; MÖLLER, D.P.F.; KESPER, B. (1999): *Realisierung und Einsatzmöglichkeiten des Geoinformationssystems BAGIS.* In: W. Lempp, G. Reik (Hrsg.): Berichte von der der 12. Nationalen Tagung für Ingenieurgeologie (Halle/Saale), preprint.

MORPHING AS PART OF A VIRTUAL REALITY FRAMEWORK FOR SURFACE RECONSTRUCTION

Dietmar P. F. Möller[1] and Björn Kesper[2]
[1]Chair Computer Engineering
& McLeod Institute of Simulation Sciences, German Chapter at University of Hamburg
Email: dietmar.moeller@informatik.uni-hamburg.de
[2]Email: kesper@informatik.uni-hamburg.de
University of Hamburg, Department Computer Science
Vogt-Kölln-Str. 30, D-22527 Hamburg, Germany

Abstract. We present the methodology of virtual reality, as a framework for surface reconstruction based on morphing mechanisms. Due to that point our claim is that our work should give insight into realized complex research case study examples showing the power of deformable models, as part of morphing, in a virtual reality framework in the spacious areas of the medical application domains. Based on that facts we will give case study examples of virtual reality applications in the different fields of applied surgery. Currently we are working on other applications of morphing in medicine.

1. Virtual Reality Methodology in the Area of Medicine

Applying the virtual reality methodology to the medical domain could be stated as combining distributed virtual environment, in order to support collaboration among team members working with space distance, developing plans and procedures, doing measurements and data processing in surgical procedures, medical research projects, clinical oriented support systems development and evaluation etc. in order to attempt to manage new investigations and organizations in a collaboratively manner, as it is needed in global as well as international project development.

One of the most interesting new paradigms in virtual reality methodology in this domain is that three dimensional representations are not only the lonely possibility of a setting.

Many virtual applications in medicine, if not already now, will in future make use of specific graphics. The virtual space will be visualized in space, which means in terms of three dimensions, and time. People in charge with virtual reality in the medical domain are able to interact with space and time, e.g. like walking through the vascular bed for inspection of collageneous settings at the vessels intima, or interacting with other medical disciplines for consultancy through a graphical user interface in the manner of computer supported cooperative work, as well as designing the plastical view of cosmetic surgery. The interweaving of functionality, distribution, efficiency, and openness aspects is very noticeable in computer graphics. The virtual space is graphically visualized flamboyance and for the most part the people in charge with the medical virtual space application domain should see the same image.

Therefore, for virtual reality applications, a three-dimensional, multi-user virtual reality tool for the medical application domain as been developed, consisting of the following main components:

- space ball and cyber gloves for tactile interaction in virtual space
- head mounted devices for visual interaction in virtual space
- 3-dimensional geometric body creation and motion methodology for "virtual space feeling" capability
- 3-dimensional visual interactive system for definition, manipulation, animation and performance analysis of medical geometric bodies
- object oriented data base for efficient data management in virtual reality applications
- hardware for the power of computing in space and time
- objects organization into inheritance hierarchies for virtual reality system transparency

When medical objects are created, they inherit the proper-ties and verbs of their ancestors. Additional verbs and properties as well as specializations of inherited components may be defined to give the new object its unique behavior and appearance.

Based on that assumptions a virtual reality simulator for the medical application domain has been build up.

2. Visualization as the basis for Morphing in Medicine

The presentation of process states is of importance, which has to be realized time dependent, bringing together real scenarios as well as virtual scenarios of the medical project under realisation as real research project, in order to find out e.g. optimal geometries, based on Non Uniform Rational B-Splines (NURBS).

This special kind of B-Spline representation is based on a grid of defining points $P_{i,j}$, which is approximated through bi-cubic parameterized analytical functions.

$$P_{i,j} = \begin{Bmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,n} \\ p_{2,1} & p_{2,2} & \cdots & p_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ p_{m,1} & p_{m,2} & \cdots & p_{m,n} \end{Bmatrix}, p_{i,j} = (x, y, z)$$

$$S(u,v) = \frac{\sum_{i=0}^{n}\sum_{j=0}^{m} N_{i,p}(u) N_{j,q}(v) w_{i,j} P_{i,j}}{\sum_{i=0}^{n}\sum_{j=0}^{m} N_{i,p}(u) N_{j,q}(v) w_{i,j}}$$

$$0 \le u, v \le 1$$

This method allows to calculate the resulting surface or curve points by varying two (surface) or one (curve) parameter values u and v of the interval [0,1], respectively, and evaluating the corresponding B-Spline basis function $N_{i,p}$.

$$N_{i,0}(u) = \begin{cases} 1 & \text{if } u_i \le u \le u_{i+1} \\ 0 & \text{otherwise} \end{cases},$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u)$$

$$U = \{u_o, \ldots, u_m\}, u_i \le u_{i+1},$$

V analogous

As the parameter values u and v can be chosen continuous, the resulting object is mathematically defined in any point, thus showing no irregularities or breaks.

There are several parameters to adjust the approximation of the given points and thus changing the look of the described medical object, so that if needed interpolation of all points can be achieved.

First of all, the polynomial order describes the curvature of the resulting surface or curve, giving the mathematical function a higher level of flexibility. Second, the defining points can be weighted according to their dominance in respect to the other control points. A higher weighted point influences the direction of the surface or curve more than a lower weighted. Further more, knot vectors U and V define the local or global influence of control points, so that every calculated point is defined by smaller or greater arrays of points, resulting in local or global deformations, respectively.

NURBS are easy to use, as modeling and especially modifying is achieved by means of control point movement, letting the user adjust the object by simply pulling or pushing the control points (Figure 1).

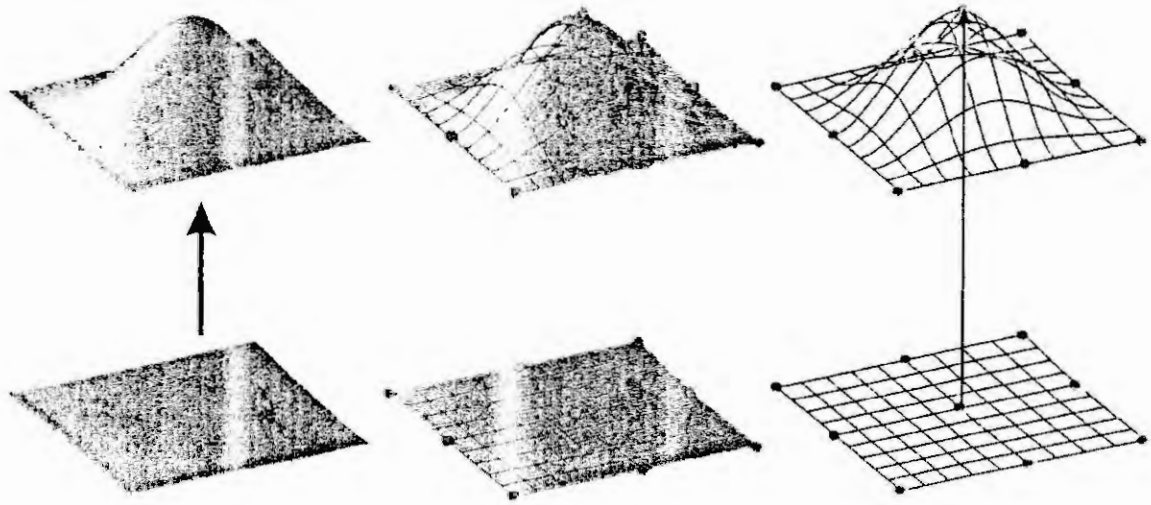


Fig. 1 Modelling and modification of a NURBS surface

Based on these concepts a methodology to interpolate a given set of points, for example the results of scanned data of humans face surface measurements, has been developed. As shown in Figure 2, huge sets of scattered data points are used to generate the resulting object.

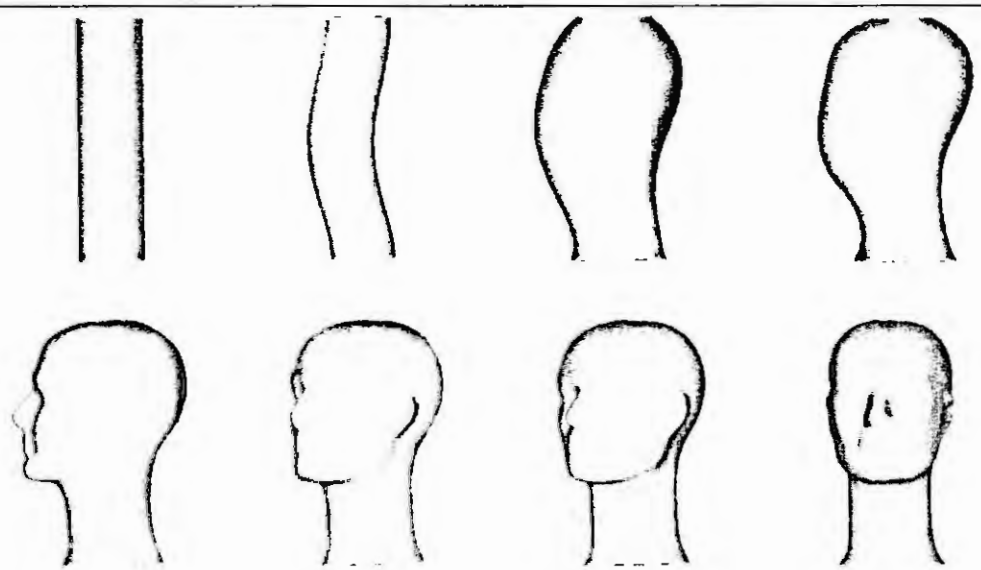


Fig. 2 Multi level B-Spline approximation

Using multiple levels of surface morphing, this multi level B-Spline Approximaton (MBA) adjusts a predefined surface, i.e. a flat square or a cylinder. Constraints like the curvature or direction at special points can be given and are evaluated within the algorithm.

Based on OpenGL, a quasi standard for three dimensional modeling and visualization, we are able to create geometric medical bodies of every shape and size and move them in real time.

3. Deformable Models in Medical Surface Reconstruction

Mathematically geometric (medical) subjects can be interpreted as embedded contour within an image plane

$$(x,y) \in \Re^2$$

of a virtual reality framework concept. The contour itself can be assumed as

$$\exists (s) = (x(s), y(s))^T$$

where x and y are the coordinate functions and $s \in [0,1]$ the parametric domain. The shape of a contour subject to an image I (x,y) can be described [McInerney et al., 1999] by the functional

$$\Im (\exists) = E (\exists) + \Gamma (\exists)$$

The functional given above can be interpreted as representation of the energy of the medical contour. Hence the final shape of this medical contour corresponds to a minimum of energy. Due to that the first term of the functional given above can be introduced as internal deformation energy

$$\Xi (\exists) = {}_0 \int^1 \Lambda_1 (s) \mid \delta\exists/\delta s \mid^2 + \Lambda (s) \mid \delta\exists^2/\delta s^2 \mid^2 \delta s.$$

This equation describes the deformation of a stretchy, and flexible medical contour, with $\Lambda_1(s)$ as tension of the medical contour and Λ_2 (s) as rigidity.

In accordance with the calculus of variations, the medical contour $\exists (s)$, which minimizes the energy $\Im (\exists)$ must satisfy the Euler-Lagrange equation [McInerney et al., 1999]

$$- \delta/\delta s (w_1 \cdot \delta\exists/\delta s) + \delta^2/\delta s^2 (w_2 \cdot \delta^2\exists/\delta s^2) + \nabla P(\exists(s,t))=0$$

The vector partial differential equation, introduced above, describe the balance of internal and external forces when the medical contour rests at equilibrium. Therefore the first two represent the internal stretching and bending forces respectively, while the third term represents the external forces that couple the contour to the image data.

4. Conclusions

The potential of virtual reality is huge. We only scratched the surface of the complex due to the medical application domain we are reporting in our case study approaches. The potentially of morphing contains an incredible number of solutions to different problem depending domains.

5. References

1. Crilly, A.J., Earnshaw, R.A. and Jones, H., Applications of Fractals and Chaos. Springer Verlag, Berlin, 1993.
2. Encarnacao, J.L., Peitgen, H.-O., Saka, G. and Englert, G., Fractal Geometry and Computer Graphics, Springer Verlag, Berlin, 1991.
3. Gilfillan, L. and Harbison, K., Using distributed virtual environments (DVE) for collaborative program planning and management: Problems and potential. In: Proc. VWSIM´98 (Eds.: Landauer, C. and Bellman, K.L.), SCS Publishers, San Diego, 1998, 39-46
4. Möller, D.P.F., Virtual Reality: Simulation Synergy in Laboratories and Outer Space Domains. In: Simualtion: Past, Present and Future (Eds.: Zobel, R. and Möller, D.P.F.), Vol. II, SCS Publishers, Delft, 1998, 64-66
5. Schneider, M., Spatial Data Types for Database Systems. Springer Verlag, Berlin, 1997.
6. Singh, A., Goldgof, D. and Terzopoulos, D., Deformable Models in Medical Image Analysis, IEEE Press, Los Alamitos, USA, 1998.
7. Straßer, W. and Seidel, H.-P., Theory and Practice of Geometric Modeling, Springer Verlag, Berlin, 1989.
8. Yachik, T.R., Synthetic Scene Quality Assessment Metrics Development Considerations. In: Proc. VWSIM´98 (Eds.: Landauer, C. and Bellman, K.L.), SCS Publishers, San Diego, 1998, 47-57

APPLICATION OF THE PROCESS MODELING TOOL PROMOT TO THE MODELING OF METABOLIC NETWORKS

M. Ginkel[1], A. Kremling[1], F. Tränkle[2], E. D. Gilles[1,3], and M. Zeitz[3]

[1] Max-Planck-Institut für Dynamik komplexer technischer Systeme,
Zenit-Gebäude Leipziger Str. 44, D-39120 Magdeburg, Phone: +49 391 6117 563,
E-mail: ginkel@mpi-magdeburg.mpg.de
[2] ETAS GmbH & Co. KG, Stuttgart, Germany.
[3] Institut für Systemdynamik und Regelungstechnik, Universität Stuttgart, Germany.

Abstract. The process modeling tool PROMOT [6, 5] has been developed for the object-oriented and equation-based modeling of chemical processes. This contribution presents the application of PROMOT to a novel application field: the design and implementation of a knowledge base for modeling metabolic networks in living cells. This knowledge base contains predefined modeling entities for the hierarchical aggregation of complex reaction network models. The principles of defining primitive metabolic entities as well as the construction of a flexible knowledge base for modeling larger metabolic networks are presented.

Key words: computer-aided process modeling, knowledge base design, object-oriented modeling language, process modeling tool, metabolic processes

1 Introduction

Today commercially and freely available software tools for the steady-state and dynamic simulation of chemical and biological processes are well established in the process engineering community. Whereas these simulation tools provide state-of-the-art numerical methods, the computer-aided development, implementation, and reuse of mathematical models for chemical and bioengineering processes are not sufficiently supported. The process modeling tool PROMOT has been designed for the computer-aided modeling of chemical processes as well as for the implementation of knowledge bases that contain reusable modeling entities [5, 6]. The differential-algebraic process models created with PROMOT are added to the model library of the simulation environment DIVA [2] by calling the DIVA code generator. The numerical methods provided by DIVA may be applied to the numerical analysis, dynamic and steady state simulation, and optimization of the process models.

2 Modeling Concept

The modeling concept of PROMOT is both based on the rigorous modeling of lumped parameter systems as well as the mechanisms of object-oriented modeling including abstraction, encapsulation, aggregation, and inheritance. In PROMOT, process models are built by aggregating structural and behavioural modeling entities that represent tbe topological structure or the dynamic and steady-state behaviour of the investigated process, respectively. Two types of structural modeling entities are distinguished: modules and terminals. Modules represent differential-algebraic process models and their submodels. Examples for modules in process engineering are models for process units (e. g. distillation columns), control volumes (e. g. thermodynamic liquid and gas phases), compartments (e. g. distillation column trays and sections), and signal transformers (e. g. control devices).

For defining a module, it must be separated from its environment. The system boundary of a module is defined by attaching terminals. Via these terminals, the module may exchange material, momentum, energy, and information with other modules. There are two types of behavioural modeling entities: process variables and ordinary differential as well as algebraic equations. The aggregation of process variables and equations leads to a differential-algebraic model for the considered module.

All modeling entities in PROMOT are organized iu an object-oriented class hierarchy with multiple inheritance. They inherit all parts and attributes from their respective superclasses.

This modeling concept was developed for chemical processes and can be readily applied to the modeling of metabolic networks as shown in this contribution. In living cells, hundreds of enzyme catalysed reactions occur simultaneously. Moreover, a large number of feedback and feedforward control loops were discovered in biochemical research. However, cells are able to respond very quickly to changes

of the environmental conditions by turning on or off metabolic pathways. It can be concluded that a powerful but sensitive control management with well balanced actions upon metabolic fluxes is realized within the cellular interior. Therefore, in microbiology, the thinking in units (describing a subset of metabolic processes) has become very popular and has resulted in the definition of subnetworks that are under control of a common regulator protein [3]. This allows the cell to stop or to activate the biosynthesis of a large number of enzymes belonging to this subnetwork. In the bacterium *Escherichia coli* a very well known subnetwork for transport and metabolism of carbohydrates is the *crp* modulon with the common regulator protein CAP (catabolite activator protein). In this subnetwork more than 40 metabolic pathways are under control of this single protein which is able to bind on the DNA and can activate the transcription of the genetic information.

A decomposition of the subnetworks leads to elemental modules or submodels [1]. This corresponds to the subsystems of chemical processes. The elemental modules are substance storages, substance transformers and signal transformers. Substance storages represent two classes of substances: on one hand intermediates of the metabolism possessing no genetic information, like precursors and amino acids and on the other hand macromolecules like proteins, DNA and RNA wbich do possess genetic information. The substance storages are mathematically described by differential equations. Substance transformers connect two or more storages and therefore represent biochemical reactions. These transformers therefore have two main aspects: (1) the representation of the stoichiometric structure of the reaction and (2) the representation of the reaction kinetics as well as the influences of participating and controlling ligands (substrates, activators and inhibitors) on this reaction. These two aspects are investigated and described separately.

Since the understanding of signal transduction and processing is the key for describing the overall behaviour of cellular systems, these processes are described by the module signal transformer. A typical example for a signal transducing process is the initiation of gene expression (first step of protein biosynthesis). The activity of the activator CAP can be varied by a small molecule (cAMP). The activation of CAP and the subsequent binding on the DNA binding site can be seen as a signal for the synthesis of the enzymes.

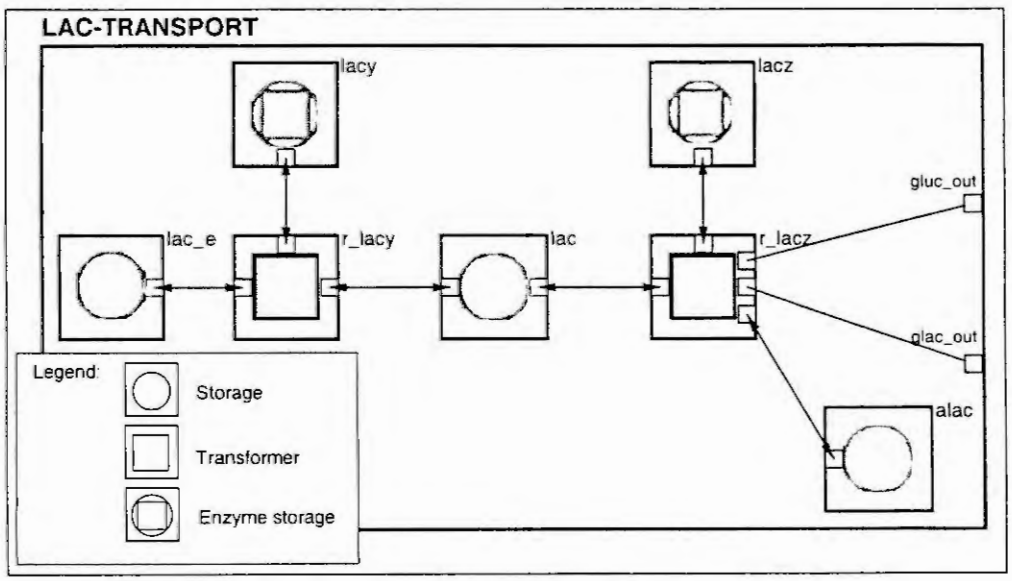


Figure 1: Structure of a simple subnetwork: lactose uptake and degradation. Lactose is taken up with the help of the enzyme lacy and is split into glucose and galactose by lacz. An important by-product is Allolactose.

Substance storages and substance transformers are aggregated to a higher structured unit: a metabolic pathway. A pathway describes a system of substances and reactions that are aggregated to a reaction chain. As an example, Figure 1 shows the uptake and degradation of the carbohydrate lactose. External lactose from the medium (lac_e) is taken up by the enzyme lacy. Intracellular lactose (lac) is subsequently metabolized by the enzyme lacz to glucose and galactose. These substances can be used by other sub-

networks through the terminals gluc_out and glac_out. The transformers r_lacy and r_lacz are named according to their enzymes. The production of the enzymes lacy and lacz is controlled by CAP on a superior level of the metabolic network. A by-product of this reaction is the metabolite allolactose (alac) which acts as an inducer for the lactose enzymes. This means that besides the activator CAP a further regulator protein is involved (laci – not shown in Figure 1) which guarantees that the enzymes responsible for lactose degradation are synthesized only when lactose is present in the medium.

3 Knowledge Base for Metabolic Models

The knowledge base is designed in a bottom-up approach. At first, elemental modules are defined, like substance storages and aspects of transformers. As mentioned above, the structure and the kinetics of transformers are represented separately by two distinct specialization hierarchies of modules in the knowledge base. The different kinds of modules can be combined by multiple inheritance in order to form many different transformers with specific reaction kinetics. E.g. the transformer TRANS2A_M1A, shown in Figure 2 is the combination of a stoichiometric structure TRANS2A with the kinetic module M1AT. This transformer is used in the example above (Figure 1) for r_lacy. The module TRANS2A represents

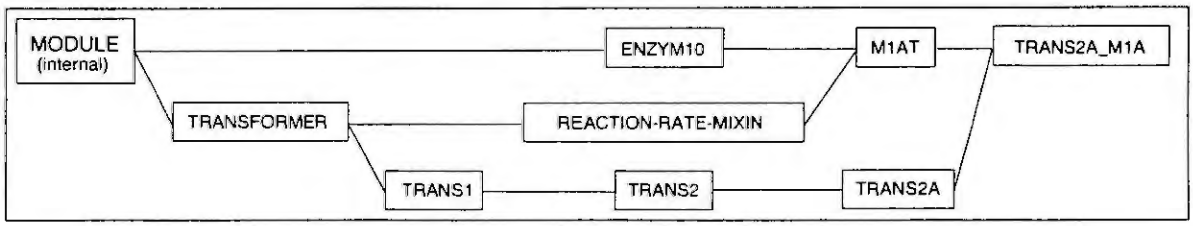


Figure 2: Inheritance of transformer TRANS2A_M1A (most special class on the right side)

a transformer for two substances with stoichiometric calculations for consumption and production of the substances while M1AT is a kinetic module which calculates the reaction rate using a Michaelis-Menten kinetic under the control of one enzyme. These two parts are abstract[1] modules that are complementing each other. TRANS2A is located in the hierarchy of structural aspects for transformers which starts with TRANS1, a drain transformer for one substance. M1AT is a reaction-rate-mixin which calculates the reaction-rate for different transformers. As it uses the concentration of one enzyme for its calculations, it includes ENZYM10 which defines a terminal and some parameters for the connection to an enzyme storage. The definition of the resulting transformer is rather simple and describes only a new name and the used superclasses. There are other tranformer modules in the knowledge base which use an identical structure but a different kinetic aspect. The proposed design of the knowledge base facilitates the integration of other kinetic aspects. Due to the stability of the interface defined by the terminals of the structural aspect module, transformer models in a larger subnetwork can be easily replaced by a different version.

Further modules in the knowledge base are composite modules that describe common metabolic pathways that are useful for different metabolisms. There are for instance modules for glycolysis, the tricarboxylic acid cycle, and for lactose uptake and degradation. With this set of modules the user can compose larger metabolic networks. Modules for these networks are defined by aggregating the predefined modules from the knowledge base and coupling them via their terminals. A complete model for a part of the metabolism of a cell that can be used in simulation is also represented by a module.

For the investigation of metabolic systems, the resulting model can be analyzed using the numeric routines of the simulation environment DIVA [5]. There are simulations performed to estimate parameters for fermentation experiments or to show the dynamic behaviour of the biological model under changing external conditions. Another important method for comprehending the interaction between the different reactions in the metabolic network is the metabolic flux analysis (MFA) [4]. MFA is based on the stoichiometric matrix of the metabolic network and does not require information about kinetic aspects. The necessary stoichiometric information for this method is already contained in the modeling entities, so PROMOT generates the input for MFA out of the same model as used for dynamic simulation.

[1] Abstract means: They cannot be used directly in a metabolic model because they are incomplete and describe only one aspect of the transformer.

Therefore special modeling entities are arranged in the specialization hierarchy that reflect the semantics of the variables and equations within the model. There are for instance modeling entities for stoichiometric parameters or intracellular substance concentrations. These entities are used throughout all the predefined modules in the knowledge base. From this formal information the correct parameters out of model are selected and arranged into the stoichiometric matrix according to the associated substances and reactions. PROMOT is now able to provide the input for a Matlab based MFA tool that is currently under development.

4 Conclusions

In this contribution, a new application field for the process modeling tool PROMOT has been presented. A modeling approach based on storages and transformers has been presented which is sufficient for the structured modeling of large metabolic networks. The modeling entities for this approach can be represented with the methodology of PROMOT and can be implemented in a flexible knowledge base by using multiple inheritance. Of great importance in this implementation process is an object-oriented analysis of the expert knowledge of the bioprocess engineer. Modeling entities in the knowledge base can be aggregated to subnetworks for common metabolic systems and complete metaholic networks.

The resulting models can be analyzed with the numerical methods of the dynamic simulation environment DIVA. Besides that a further tool for MFA has been integrated. With additional semantic information in the knowledge base, the input for this tool can be generated from the same models that are used for dynamic simulation.

Further development has to be done for the modeling of the entities for the signal transduction network. Therefore another part for the knowledge base will be implemented. During the modeling of larger subnetworks some prohlems arose due to the current rigid methodology of connections in PROMOT. Some improvements have to be done to make this methodology more flexible. Therefore, as a new concept, the stretchable terminal will be introduced, which allows an indexed vector of terminals to adjust its dimension to the number of links it has to other terminals. This will facilitate the definition of general storages and transformers with arbitrary numbers of terminals.

A further extension aims to the access of external databases for the automatic retrieval of stoichiometric and kinetic parameters of metaholic pathways. In hiological research, much data about enzymes and already known pathways for different biological species has been collected in large databases. This data is available over the internet and can he used for the automatic retrieval of parameters or in a further extension to PROMOT for the semi-automatic generation of metabolic models.

References

[1] G. Breuel, E. D. Gilles, and A. Kremling. A systematic approach to structured biological models. In *Proc. 6th Conference on Computer Applications in Biotechnology*, pages 199–204, Garmisch-Partenkirchen, Germany, 1995. Dechema.

[2] K. D. Mohl, A. Spieker, R. Köhler, E. D. Gilles, and M. Zeitz. DIVA - A simulation environment for chemical engineering applications. In *Informatics, Cybernetics and Computer Science (ICCS-97), Collected Volume of Scientific Papers*, pages 8–15. Donetsk State Technical University, Donetsk, Ukraine, 1997.

[3] F.C. Neidhardt, J.L. Ingraham, and M. Schaechter. *Physiology of the bacterial cell: A molecular approach.* Sinauer Associates, Sunderland, Massachusetts, 1990.

[4] J. Nielsen and J. Villadsen. *Bioreaction Engineering Principles.* Plenum Press New York and London, 1994.

[5] F. Tränkle. *Rechnerunterstützte Modellierung verfahrenstechnischer Prozesse für die Simulationsumgebung DIVA.* Dissertation, Universität Stuttgart, 1999. In progress.

[6] F. Tränkle, A. Gerstlauer, M. Zeitz, and E. D. Gilles. PROMOT/DIVA: A Prototype of a Process Modeling and Simulation Environment. In I. Troch and F. Breitenecker, editors, *IMACS Symposium on Mathematical Modelling, 2nd MATHMOD*, pages 341–346, TU Vienna, Austria, February 1997. ARGESIM Report No. 11.

NLMIMO, NON-LINEAR MULTI-INPUT MULTI-OUTPUT TOOLBOX

Edy Bertolissi, Antoine Duchâteau, Hugues Bersini
IRIDIA, Université Libre de Bruxelles
50, Av. Franklin Roosevelt, 1050 Bruxelles, Belgium
email: eberto@iridia.ulb.ac.be, aduchate@ulb.ac.be, bersini@ulb.ac.be

Abstract. NLMIMO, Non-Linear Multi Input Multi Output, is a MATLAB toolbox for modeling non-linear dynamic systems. The toolbox has been developed using an object oriented approach. Model representations for linear, Takagi-Sugeno, Mamdani, and lazy mappings, as well as various model identification techniques, have been integrated. They are available as a set of interchangeable classes which can be easily extended. The toolbox is completed by a series of tools for the visualization of the models, and a database management utility for listing the properties of the mappings associated to the system, comparing their performance and validating them.

1 Introduction

Learning models from observations and studying their properties is a major issue in several scientific and industrial fields. People store observed data since they believe that it is possible to extract some useful information from it. However once the data has been collected it is necessary to decide how to learn something from it. Building a model is a way for trying to capture, and mathematically define, the rules governing a process.

Since the idea of extracting useful knowledge is present in many disciplines, from econometrics to machine learning, from statistics to control, a large number of approaches have been developed to solve this problem. Least-squares linear approximators, neural networks, fuzzy systems, and lazy learning approximators, are different ways of transforming the available empirical data in a set of mathematical equations useful for the intended application. These approaches are not equivalent. They present discrepancies with respect to the class of functions they are adapted to, the approximation efficiency and the number of parameters needed to approximate a given nonlinearity. An approach can be global or local, which means that parameter adjustment can have a global or local impact on the model. Other important distinctions among the possible approaches are their computational efficiency, their transparency, that is how readable the parametrisation is, and how easy it is to introduce prior knowledge into the model.

Some approaches are better suited than others to solve certain classes of problems. NLMIMO has been designed as a toolbox which allows the user to choose, experiment and compare different approaches for modeling from data. In the case of multi-output systems it is even possible to choose a different modeling approach for each output, increasing the overall flexibility of the system. At present NLMIMO implements linear, Takagi-Sugeno [7], Mamdani [5], and lazy [1] modeling approaches. Takagi Sugeno and Mamdani models are two well known types of fuzzy models. Lazy models, also known as instance based models, defers processing of the training data until a query explicitly needs to be answered. This means that all the training data needs to be stored in memory and accessed when a query is made. Due to the structure of the toolbox, the inclusion of other approaches is a simple process.

2 Overview of NLMIMO

Raw data is rarely immediately beneficial, but its true value derives from the possibility of extracting information which could be used for understanding the process which generated the data, and predicting its future values. In many cases previous knowledge about the process is advantageous for better treating the observed data. This means that the process for identifying a model is not sequential but iterative: it implies feedback from the user and interactions where the model is refined until it provides a good description of the phenomenon underlying the data. During the refinement procedure, the user must always keep into mind the objective of the modeling procedure, the context in which the model will be used. For example a model suited for prediction is not always adapted to control.

The main scope behind NLMIMO is the integration of the software produced by the partners taking part in the FAMIMO project (ESPRIT LTR Project 21911) in a unique toolbox. Therefore it has been necessary to design a flexible system that can accommodate the needs of the different laboratories collaborating to the project.The use of an object oriented approach, which allows modularity, flexibility

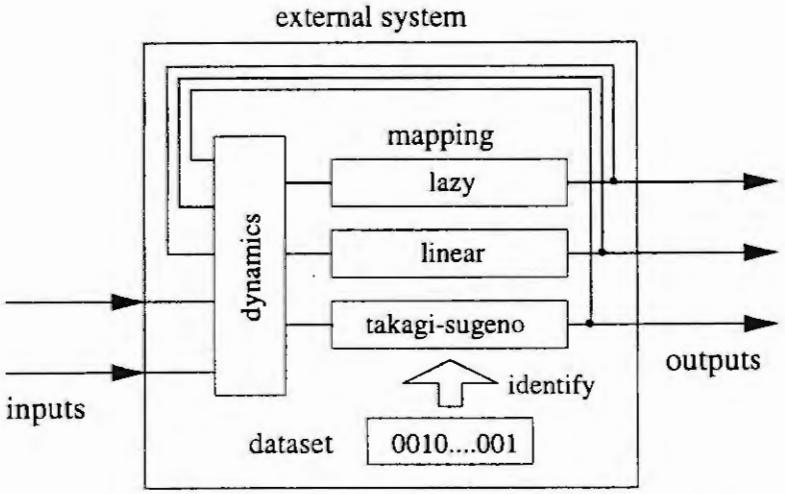


Figure 1: Relationship among the classes defined in the NLMIMO toolbox

and ease of extension had been a natural choice for the design of the toolbox. This approach has been possible thanks to the new object oriented features introduced starting from MATLAB 5.0.

The NLMIMO toolbox is built around three main classes. The `system` class, which defines all the attributes which characterize a system, the `dataset` class, which allows the storage and the processing of the data, and the `mapping` class, which defines the modeling of the process. The `system` class is the main class of the toolbox, and the other ones are used to describe some particular features of the system. In this way the data and the parameterization of the model are separate from the description of the system, and therefore they can be easily modified or updated without the need of altering the structure of the `system` class. Each class has methods for accessing, setting and displaying its attributes, and checking its integrity and consistency.

The `dataset` class is used to handle the data. Once an object belonging to this class is initialized with raw data, several methods for performing data manipulation (e.g. filtering) and data analysis (e.g. clustering) are implemented.

The `mapping` class is used to define an inputs/outputs relation. Since it is possible to choose among different possible alternatives for defining the mapping between a specified number of inputs and outputs, the `mapping` abstract class, defines all the features which are common to all the possible mappings. The definition of the details, which are dependent from the particular descriptor, are left to its subclasses. Virtually every type of non-linear function can be implemented as a subclass of the `mapping` class. At present linear systems, Mamdani fuzzy systems, Takagi-Sugeno fuzzy systems, and lazy learning systems (integrated from [4]) are available in the toolbox. New mappings can be easily added by defining other subclasses of the `mapping` class. With the exception of the linear models, all the mappings may be used to define nonlinearities. Since they have been shown to be universal approximators for certain classes of functions, it means that they are equivalent with respect to the nonlinearity which they approximate. Even if from the process point of view it is the nonlinearity that matters, not how it is parametrized, from the point of view of the designer the parameterization selected can be very important.

The `system` class centralizes the information needed to describe a general static or dynamic system. Since a system can be represented using an internal (state space based) or external (input-output based) representation, two subclasses, the `external` and the `internal` one, have been defined to implement these two alternative descriptions. Figure 1 shows a graphical representation of a 2 inputs 3 outputs system belonging to the `external` class. The object stores the data associated to the system which is used for the identification of the 3 mappings which define the input-output relations. It possible to see that a different type of mapping is used for modeling each output. The structure of this class allows a very general representation of dynamical systems, and lets the user experiment and mix different types and sizes of models.

3 NLMIMO functionalities

The NLMIMO toolbox offers to the user all the functionalities necessary to carry out a complete modeling of the system starting from data file. This can be done building the required objects, and applying the appropriate methods entering the appropriate commands at the MATLAB command line. An alternative approach consists in taking advantage of the NLMIMO graphical user interface (GUI), which has being designed to help the user to carry out the modeling process in a structured way.

The identification process of the model of the manifold pressure of a direct injection engine will be used to describe the functionalities of the toolbox. The engine has two inputs, the fresh air throttle control and the engine speed, and one output, the manifold pressure. A sequence of 2000 input-output pairs is available for identification purposes. First of all it is necessary to define the engine as an object belonging to the `external` class. Then it is necessary to specify the limits of the variables, the sampling time and dynamics of the system. Finally the training data is attached to the object, and the parameter identification procedure can start.

Parameter identification methods

Parameter identification is a process which consists in determining the parameters associated with the selected mapping. This is a procedure which is highly dependent of the type of chosen mapping, and the features to optimize.

Not all the mappings implemented in the toolbox require this step. Lazy learning modeling does not need parameter identification, since the local model is built on query-by-query basis. The whole parameter identification process consists in manually selecting the dimension of the regressor, defining the minimum and maximum number of neighbors considered in each query and choosing the type of approximator. Similarly for the Mamdani fuzzy systems there is no parameter identification step, since these models are usually defined starting from the knowledge of an expert about the process. The linguistic description of the process is used to build the model, which is then manually tuned to maximize its performance.

The other mappings implemented in the toolbox define methods which, once defined some mapping dependent constraints, automatically determine the parameters associated with the selected mapping. Linear models use the least mean square method to find the parameters associated to the model. Takagi Sugeno mappings offer a wide choice of identification methods: identification by product-space clustering (`fmclust` [2], `cluslms`, `cluslev`), incremental identification (`incrsie` [6], `tsgklmsxv` [3]), and global identification (`randlm`).

It is possible to use different mappings and choose alternative identification procedures in order to identify the model of the manifold pressure system. The left part of figure 2 shows the NLMIMO visualization tool after the system has been identified using five different identification procedures (listed in the top left corner of the image). Even if in this case only one identified model has been selected for visualization, it is possible to choose at the same time multiple mappings, and visually compare their shapes, errors and other interesting features.

Model validation methods

Model validation, the process which leads to the model selection, is application specific. It is often the result of a compromise between different factors, such as the quantitative measurements, the personal experience of the designer and the effort needed to implement a particular model in practice. The NLMIMO toolbox provides a series of functionalities for evaluating quantitative criteria commonly used in validation techniques. An error database allows to access to the simulation and prediction results of the different mappings, visualizing and comparing them. The results of other statistical criteria such as estimated prediction error and the results of the cross validation can be analyzed and compared.

The right part of figure 2 displays the error database of the manifold pressure which stores the mean square error of the different mappings. In this particular case it is possible to see that the identification methods based on the Takagi-Sugeno fuzzy system provide comparable results both on the training and validation set. The performance of the lazy mapping is inferior, mainly because the number of training points is quite low in comparison with the complexity of the process. Without surprises the linear

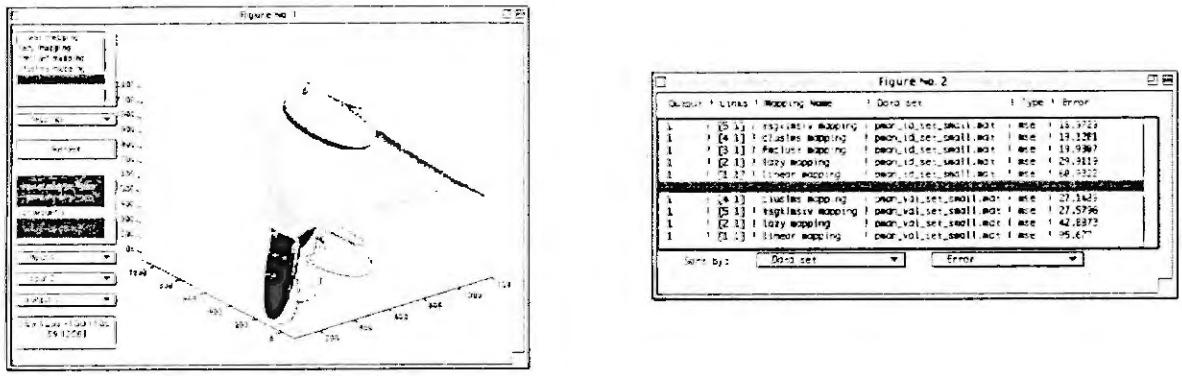


Figure 2: The model viewer (left) allows to visualize the models generated by the different mappings associated to the system, while the model error database (right) displays the list of all means squares error of the mappings of the system.

mapping gives very poor results. A similar database exists for storing the results of the means square simulation errors which are calculated using the input data and using the outputs calculated by the model to build the new regressor. This means that errors accumulate during the simulation, which let us check if the model really captures the dynamic of the system . The simulation of the models can be displayed to compare the performance of the mappings.

4 Conclusion

The NLMIMO toolbox implements several state of the art techniques used in multimodel based modeling and identification processes. Even if the modeling aspects of the toolbox are the predominant ones, support for the development of controllers and the study of their stability is also included in the toolbox. One of the main targets of this toolbox has been to design a flexible set of utilities which allow the user to experiment and compare several solutions. Even if computational efficiency was an important feature in the design of the toolbox, it was not the main issue. If a particularly performant system is required, the best approach is to reimplement the relevant functions once the best method has been identified. Future works include the integration of other identification techniques such as neural networks and mixture of experts in the toolbox, and the release this toolbox to the scientific community on the Internet.

References

[1] D. W. Aha. Editorial. *Artificial Intelligence Review*, 11(1–5):1–6, 1997.

[2] R. Babuska. *Fuzzy Modeling and Identification.* PhD thesis, Technische Universiteit Delft, 1996.

[3] H. Bersini, A. Duchateau, and N. Bradshaw. Using incremenmtal learning algorithms in the search for minimal and effective fuzzy models. In *Proceedings of the FUZZ-IEEE '97*, pages 1417–1422, Barcelona, Spain, 1997.

[4] M. Birattari and G. Bontempi. The lazy learning toolbox, for use with matlab. Technical Report TR/IRIDIA/99-7, IRIDIA-ULB, Brussels, Belgium, 1999.

[5] E. H. Mamdani. Application of fuzzy algorithm for control simple dynamic plant. *Proc. IEEE*, 121(12):1585–1588, 1974.

[6] D. Passaquay, P. Bortolet, S. Boverie, and A. Titli. Iterative fuzzy modelling of non-linear functions, application for the control of non-linear systems. In *Proceedings of the European Control Conference, ECC'99*, Karlsruhe, Germany, 1999.

[7] T. Takagi and M. Sugeno. Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetics*, 15(1):116–132, 1985.

SIMURV. A SIMULATION PACKAGE FOR UNDERWATER VEHICLE-MANIPULATOR SYSTEMS

Gianluca Antonelli

Dipartimento di Informatica e Sistemistica
Università degli Studi di Napoli Federico II
Via Claudio 21, 80125 Napoli, Italy
http://disna.dis.unina.it/prisma
antonell@unina.it

Stefano Chiaverini

Dipartimento di Automazione, Elettromagnetismo,
Ingegneria dell'Informazione e Matematica Industriale
Università degli Studi di Cassino
Via G. Di Biasio 43, 03043 Cassino (FR), Italy
chiaverini@unicas.it

Abstract. In this paper a software package for the simulation of the direct dynamics of Underwater Vehicle-Manipulator Systems (UVMSs) is presented. The software, based on the MATLAB-SIMULINK environment, is designed so as to allow the user to test kinematic, dynamic and interaction control laws. Two algorithms are implemented, the Articulated Body (AB) and the Composite Rigid Body (CRB), which both avoid the need of computing the overall symbolic model. The main dynamic effects have been taken into account: namely, inertial generalized forces, hydrodynamic effects, thrusters dynamics, sensor characteristics, and interaction of the manipulator end effector with the environment. A library of forward kinematics functions is available to compute, during the simulation, the end-effector position or the Jacobian of the generic simulated system. An user interface has been developed to simplify the implementation of control laws.

Introduction

The use of computer simulations is of self-evident importance when dealing with complex dynamic systems such as Underwater Vehicle-Manipulator Systems (UVMSs). Simulation tests allow to gain better understanding of the system's dynamics which can be exploited to improve the control system design. This also avoids costs and risks of experimental tests in a preliminary stage of the system design.

The basic functionality to be implemented in a computer simulator for UVMSs is the computation of the *direct dynamics*, i.e., computation of the system's accelerations, for given positions, velocities and input torques (see Figure 1). In the case of simple robotic systems it is possible to derive the symbolic dynamic model and to numerically compute the actual accelerations. Referring to Autonomous Underwater Vehicles (AUVs), several simulators have been proposed; in this case, however, the symbolic expression of the model is available [5]. With respect to UVMSs, an efficient simulator has been proposed in [7].

On the other hand, the symbolic model of a complex structure, such as an UVMS, is very difficult to derive in closed form and its expression is usually untractable. Efficient approaches to compute the direct dynamics of these systems are instead based on recursive numerical algorithms which chain the dynamics of the single bodies constituting the overall system. An additional advantage of such numerical algorithms is the possibility of easily matching different systems by changing the parameters of the single bodies and the constraints between connected bodies; in this way is very simple to change the mechanical structure and the number of bodies in the system. Two main algorithms have been used in the literature to simulate chained-body robotic systems: namely, the Composite Rigid Body (CRB) method [2, 6, 8], and the Articulated Body (AB) method [3, 7].

In this paper a software package for dynamic simulation of UVMSs is presented. The software, based on the popular MATLAB-SIMULINK environment, has been designed so as to allow the user to test kinematic, dynamic and interaction control laws. The overall system dynamics is obtained by chaining the dynamics of individual rigid bodies in a fluid environment; therefore, the user can easily modify the kinematic as well as dynamic structure of the system, e.g., number of links, Denavit-Hartenberg parameters, dynamic parameters, hydrodynamic effects, etc. Several effects are taken into account in the system dynamics: namely, inertial generalized forces (i.e., forces, moments, and joint torques), lift and drag generalized forces, ocean current loads (Munk moment), buoyancy, motor joint dry and viscous friction, thrusters dynamics, noise affecting sensor readings, and interaction of the end effector of the manipulator with the environment. A library of forward kinematics functions is available to compute, during the simulation, the end-effector position or the Jacobian of the generic simulated system.

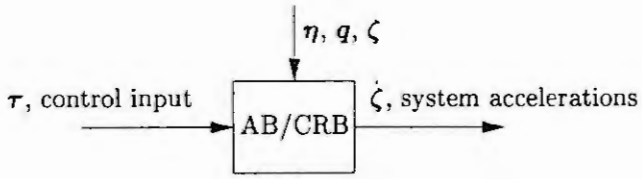


Figure 1: Direct dynamics.

Algorithm	mul/div	add/sub
CRB	$0.16n^3 + 11.5n^2 + 160.33n - 109$	$0.16n^3 + 7n^2 + 138.83n - 102$
AB	$300n - 267$	$279n - 259$

Table 1: Computational load for the two algorithms.

In order to avoid the possible occurrence of representation singularities that can arise by the use of minimal representations of the orientation, e.g., using Euler angles, a non minimal representation, based on the unit quaternion, has been used to represent the orientation of all the rigid bodies of the system.

Both the AB and CRB algorithms have been implemented in view of the different computational load and for cross checking purposes. To decrease simulation time the simulator has been coded using Mex-files.

Modelling

The equations of motion of the UVMS in a body-fixed reference frame can be written in the form [9]

$$\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{\zeta}} + \boldsymbol{C}(\boldsymbol{q},\boldsymbol{\zeta})\boldsymbol{\zeta} + \boldsymbol{D}(\boldsymbol{q},\boldsymbol{\zeta})\boldsymbol{\zeta} + \boldsymbol{g}(\boldsymbol{q},\boldsymbol{R}_I^B) = \boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{\zeta}} + \boldsymbol{n}(\boldsymbol{q},\boldsymbol{R}_I^B) = \boldsymbol{\tau} + \boldsymbol{J}_w^{\mathrm{T}}(\boldsymbol{q})\boldsymbol{h}, \tag{1}$$

where $\boldsymbol{\zeta} = [\boldsymbol{\nu}^{\mathrm{T}}\ \dot{\boldsymbol{q}}^{\mathrm{T}}]^{\mathrm{T}}$, $\boldsymbol{\nu} \in \mathbb{R}^6$ is the vector of vehicle linear and angular velocities expressed in the body-fixed reference frame, $\boldsymbol{R}_I^B \in \mathbb{R}^{3\times3}$ is the rotation matrix from the inertial frame to the body-fixed frame, $\boldsymbol{q} \in \mathbb{R}^n$ is the vector of joint variables, n is the number of joints, $\boldsymbol{M}(\boldsymbol{q}) \in \mathbb{R}^{(6+n)\times(6+n)}$ is the inertia matrix of the UVMS, $\boldsymbol{C}(\boldsymbol{q},\boldsymbol{\zeta})$ is the matrix of Coriolis and centripetal generalized forces, $\boldsymbol{D}(\boldsymbol{q},\boldsymbol{\zeta})$ is the matrix of friction and hydrodynamic damping terms, $\boldsymbol{g}(\boldsymbol{q},\boldsymbol{R}_I^B)$ is the vector of restoring forces, $\boldsymbol{\tau} \in \mathbb{R}^{6+n}$ is the vector of forces and moments acting on the vehicle and joint torques, $\boldsymbol{J}_w(\boldsymbol{q}) \in \mathbb{R}^{6\times(6+n)}$ is a Jacobian matrix defined in (2), and $\boldsymbol{h} \in \mathbb{R}^6$ is the vector of the contact force/moment at the end effector. It is also useful to define $\boldsymbol{\eta} \in \mathbb{R}^6$ as the vector of vehicle position and Euler angles in a earth-fixed reference frame.

The end-effector velocities (expressed in the inertial frame) are related to the body-fixed system velocity by a suitable Jacobian matrix, i.e.,

$$\dot{\boldsymbol{\eta}}_{ee} = \boldsymbol{J}_w(\boldsymbol{R}_B^I, \boldsymbol{q})\boldsymbol{\zeta}. \tag{2}$$

In the simulator, some simplified relationships for the hydrodynamic effects of the manipulator have been implemented [9], while a more detailed description has been used for the vehicle [5].

Description of the algorithms

Both the AB and CRB algorithm compute the accelerations of the system, subject to a specific driving torque, without resorting to the symbolic model. For a submerged body, in fact, the symbolic expression of its equations of motion is characterized by a large number of dynamic parameters [4]. Considering an UVMS, i.e., a serial chain of rigid bodies, yields to a set of equations that is difficult to compute in closed form.

The two algorithms have a different computational load; in particular, the AB is $O(n)$ while the CRB is $O(n^3)$. However, since the latter has smaller coefficients it is convenient to resort to the one or the other depending on the degrees of freedom of the structure to be simulated. Table 1 reports the computational load for the two algorithms considered for a ground fixed manipulator with n links and without taking into account the trigonometric functions.

The AB algorithm has been presented in several papers [3, 7]. In this paper, only the basic idea of this approach is given.

The AB algorithm is based on the following steps:

1. forward recursion (from the vehicle to the end effector) to compute the positions and velocities of the rigid bodies;
2. backward recursion to compute $\boldsymbol{n}(\boldsymbol{q}, \boldsymbol{R}_I^B)$ and the articulated inertias *felt* at each joint;
3. forward recursion to compute the accelerations.

The CRB algorithm is based on the computation of the inverse dynamics several time for each simulation step in order to obtain the numerical representation of the inertia matrix $\boldsymbol{M}$ and invert it to compute the accelerations [2, 6, 8].

The following steps are necessary:

1. forward recursion (from the vehicle to the end effector) to compute the positions and velocities of the rigid bodies;
2. backward recursion to compute $\boldsymbol{n}(\boldsymbol{q}, \boldsymbol{R}_I^B)$;
3. backward recursion considering only the inertial terms of $\boldsymbol{M}(\boldsymbol{q})$ with $\dot{\boldsymbol{\zeta}}$ equal zero except for the i-th element, this gives the i-th column of matrix $\boldsymbol{M}$;
4. repeat point 3 for all the degrees of freedom in order to compute the numerical value of $\boldsymbol{M}(\boldsymbol{q})$;
5. compute $\dot{\boldsymbol{\zeta}}$ by: $\dot{\boldsymbol{\zeta}} = \boldsymbol{M}^{-1}(\boldsymbol{q})(\boldsymbol{\tau} - \boldsymbol{n}(\boldsymbol{q}, \boldsymbol{R}_I^B))$.

SIMURV

A sketch of the SIMULINK model is given in Figure 2. The software package has been written in the versions MATLAB 5.2 and SIMULINK 2.0. Both the algorithms have been coded using the Mex-files to decrease the execution time. The execution time, of course, depends on the machine and the selected dynamic terms. On a PC Pentium MMX, 233 MHz, 64 MB RAM, the execution time for 1 simulation step of a 6-dof manipulator mounted on a 6-dof vehicle with all the dynamic terms is about 5 ms for the AB algorithm and about 8 ms for the CRB algorithm. It must be noted that this comparison is only indicative since the forward kinematic computations have not been optimized.

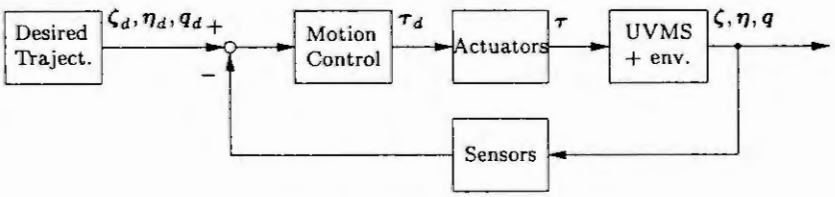


Figure 2: Simulink model.

The *Desired Trajectory* block is provided in two version. If the trajectory is given in terms of task space variables, an inverse kinematics algorithm is provided in order to output the body-fixed variables ζ_d, η_d, q_d. The *Motion Control* block can be developed by the user. Different control laws, e.g., [1], have been tested and implemented, as a demo, in the program. The *Actuator* block is based on the thrusters dynamic model presented in [10]. The *UVMS + env.* block represents the core of the program. The interaction with the environment, the presence of the hydrodynamic terms, the gravity and the buoyancy can be selected by the use of suitably defined flags. This allows the user to simulate underwater vehicle-manipulator systems as well as satellite-manipulator systems. Moreover, the influence of the different terms can be easily outlined by running two simulations with different flags. The *Sensor* block is designed to simulate the sensor behavior. A discretization and a quantization are performed, and zero mean Gaussian noise is added to the measured variables.

The simulation run is based on the following steps:

1. Collect all the data in an ASCII file. The data are naturally grouped in: simulation data (sampling time, total duration, ...); kinematic parameters data (number of links, kinematic structure of the manipulator, ...); dynamic parameters (masses, centers of gravity/buoyancy, hydrodynamic terms, ...); control law parameters.
2. Write a control law in MATLAB/SIMULINK syntax.
3. Run the simulation, choose a data file, the controller file and the output file; all the files used in the simulations, together with the output data, are saved in the working directory with suitable names.

A library of models is also available. In Figure 3 the first menu's window is shown. The user can choose between some models already defined. In the implemented models, the vehicle is modelled as described in [5]. The model corresponding to a 6-link manipulator mounted under the vehicle is obtained considering the dynamic parameters of the SMART3-S manufactured by COMAU.

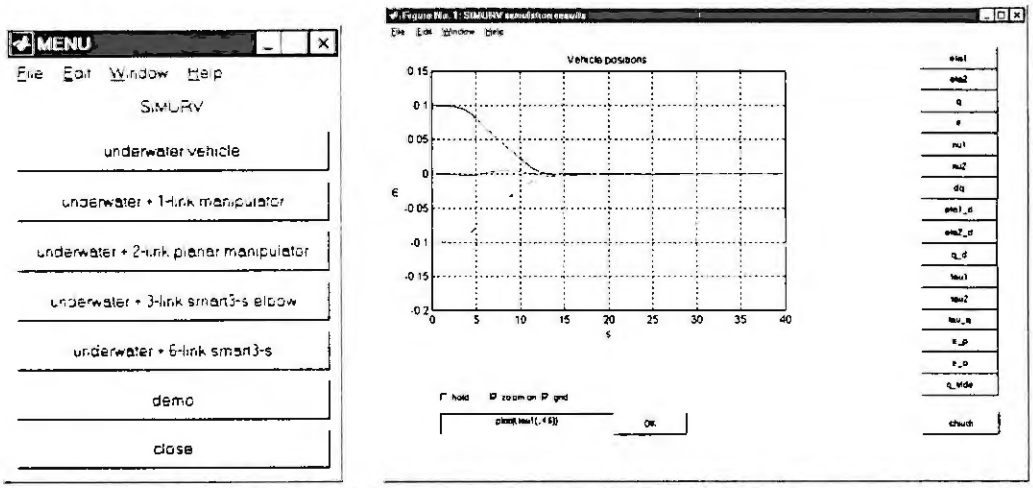


Figure 3: Left: menu of the existing models to be simulated. Right: Plot of the output results of a simulation.

By resorting to the Matlab Graphical User Interface (GUI) a window to plot the output results is opened at the end of the simulation (see Figure 3).

Summary

In this paper a software package for the simulation of the direct dynamics of serial-chain rigid bodies (in particular Underwater Vehicle-Manipulator Systems) has been presented. The software, based on the popular MATLAB-SIMULINK environment, has been designed so as to allow the user to test kinematic, dynamic and interaction control laws. Two algorithms have been implemented, the Articulated Body (AB) and the Composite Rigid Body (CRB), which do not require computation of the overall symbolic model. A library of forward kinematics functions is made available to compute, during the simulation, the end-effector position or the Jacobian of the generic simulated system. An user interface has been developed to simplify the implementation of control laws.

References

[1] G. Antonelli, F. Caccavale and S. Chiaverini, "A Modular Scheme for Adaptive Control of Underwater Vehicle-Manipulator Systems," *Proc. 1999 American Control Conference*, San Diego, CA, pp. 3008–3012, June 1999.

[2] M. Brady, J.M. Hollerbach, T.L. Johnson, T. Lozano-Perez, M.T. Mason, *Robot Motion: Planning and Control*, MIT Press, 1982.

[3] R. Featherstone, *Robot Dynamics Algorithms*, Kluwer, Boston, MA, 1987.

[4] T. Fossen, *Guidance and Control of Ocean Vehicles*, John Wiley & Sons, Chichester, UK, 1994.

[5] A.J. Healey and D. Lienard, "Multivariable Sliding Mode Control for Autonomous Diving and Steering of Unmanned Underwater Vehicles," *IEEE Journal of Oceanic Engineering*, vol. 18, pp. 327–339, 1993.

[6] W. Hooker and G. Margulies, "The Dynamical Attitude Equations for an n-Body Satellite," *Journal Astronaut Science*, vol. XII, n. 4, p. 123, 1965.

[7] S. McMillan, D.E. Orin and R.B. McGhee, "Efficient Dynamic Simulation of an Unmanned Underwater Vehicle with a Manipulator," *Proc. IEEE Int. Conf. on Robotics and Automation*, San Diego, CA, pp. 1133–1140, 1994.

[8] N.W. Walker and D.E. Orin, "Efficient Dynamic Computer Simulation of Robotic Mechanisms," *Trans. ASME—Journal of Dynamic Systems, Measurements, and Control*, vol. 104, pp. 205–211, 1982.

[9] I. Schjølberg and T. Fossen, "Modelling and Control of Underwater Vehicle-Manipulator Systems," *Proc. 3rd Conf. on Marine Craft Manoeuvring and Control*, Southampton, UK, pp. 45–57, 1994.

[10] D.R. Yoerger, J.G. Cooke, and J.-J. Slotine, "The Influence of Thruster Dynamics on Underwater Vehicle Behavior and their Incorporation into Control System Design," *IEEE Journal of Oceanic Engineering*, vol. 15, pp. 167–178, 1990.

AN INTERACTIVE RULE BASE FOR FLEXIBLE MANUFACTURING SYSTEM

I. Mariem GZARA, Slim HAMMADI and Pierre BORNE
L.A.I.L-URA CNRS- ECOLE CENTRALE DE LILLE, Tel. 33.3.20.33.54.04
BP 48, 59651 Villeneuve D'Ascq, France.
II. Sonia HAJRI
ECOLE NATIONALE D'INGENIEURS DE MONASTIR,
Route de Kairouan, 5000 Monastir, Tunisia.

Abstract This paper deals with the use of an Interactive Fuzzy Scheduling Rule Base for flexible manufacturing systems (IFSRB). We consider a rule base where each rule is balanced with a weight, which can be interpreted as a measure of the importance of this rule, compared with the others. The consequences and antecedents are both fuzzy. The antecedents of a rule are fuzzy propositions describing the scheduling environment in real time. The consequence of each rule is the priority of the operation selected by this rule. Since, all propositions do not have the same importance and influence, on the objectives of the operator; we associate to each proposition a degree of belief. The IFSRB is parametrable in order to allow the operator to react to changes. When the scheduler is dissatisfied with a decision given by the scheduling tool, a new Backward Fuzzy Reasoning Algorithm (BFRA) is developed to evaluate the degree of truth of any decision proposed by the operator.

I. Introduction

Scheduling has been examined in the operation research literature since the early fifties. It can be defined as the determination of the time sequencing of jobs and the allocation of valuable required production resources (personal, machines, tools, etc...) to accomplish the selected set of operations. An example would be Flexible Manufacturing System (FMS) where different production resources can perform a given operation.

Consequently, operations can flow in the system through different routes, so that a number of processing possibilities exists at each stage of the manufacturing process.

The scheduling problem is known to be NP-complete, i.e., the time to reach a solution grows exponentially when the size of the search space of the problem grows, whatever algorithm is used. In this paper finding the optimal solution does not interest us but we propose the use of an Interactive Fuzzy Scheduling Rule Base (IFSRB).

The IFSRB is composed by a set of fuzzy if-then rules. IFSRB is interactive, since each rule is balanced with a Certitude Factor (CF) and each proposition with a degree of belief. In fact, this base is parametrable in order to allow the user to adapt the IFSRB to changes in real time and to solve conflicts dynamically.

II. Scheduling rules

Job-shop scheduling through simulation uses various kinds of scheduling rules such us:

SPT: Shortest Processing Time,
EDD: Earliest Due Date,
MST: Minimum Slack Time.

A scheduling rule is used to select the next operation to be processed from a set of operation waiting queue. Panwalker and Iskander [12] have enumerated 113 rules and many rules continue to appear in recent works. Many researchers [1], [2], [3], [4], [13], [5] and so on have been interested by the use of scheduling rules in job shop scheduling problems. A literature review can be found in Montazari and Wassenhove [6].

Despite the important work dealing with the use of scheduling rules, very few general results are communicated [1] and [7]. Each rule aims at performing one selected criterion. However, the FMS is a multipurpose problem and generally many criteria have to be taken into account and some of them are antagonistic. A given rule can perform a selected criterion but give no efficient result according to another criterion. Furthermore, the performance of a rule on a given criterion often depends on the characteristics of the workshop, or of the set of manufacturing orders scheduled (number of jobs, number of operations, resources required, ...).

The main difficulty in FMS problems is to choose the scheduling rule useful for the system. In fact, there is no efficient rule, which is of higher performance than the others. Consequently, it is obvious to combine between rules in order to compromise between the objectives. We proposed then a new approach for FMS based on the use of a Fuzzy Scheduling Rule Base (FSRB). Each rule R_i is balanced with a certitude factor CF_i. This parameter represents the strength of belief (the importance) of the rule in comparison with the other rules in the FSRB. These certitude factors must be chosen in accordance with the objectives and with the scheduling environment.

III. Uncertainty in flexible manufacturing system

Uncertainty in the flexible-manufacturing environment contributes to the difficulty of the production problems. The production environment may be disturbed by machine breakdown, loss of materials and efficiency of individual workers such as speed, habituation and polyvalence. In this paper we extend the work of Liouane 97 in [8], [9], [10] and [11], which proposed a Fuzzy Scheduling Model (FSM) for the flexible job shop scheduling problem.

The FSM gives in real time a fuzzy representation of the state of the production environment. The processing operation time is a fuzzy variable characterised by three linguistic terms: short, middle and long.

Since different machines may execute many operations but not with the same performance, each machine is described by its fuzzy static machine activity. The machine M_k may be fast, normal or slow.

In order to avoid bottlenecks in the system, the average machine utilisation is a fuzzy variable computed in real time.

In addition, we assume that all the orders do not have the same importance in workshop. The importance of a job in comparison with the other jobs is essentially subjective. The importance of a job is a fuzzy variable (important job, no-important job).

Job must be produced in time, then each job and each operation is characterised by two fuzzy subsets urgent and no-urgent.

The production environment is described in real time by fuzzy propositions such that: "operation O_{ij} is urgent", "job J_j is important", machine M_k is slow", ...

IV. Fuzzy Scheduling Rule Base (FSRB)

Fuzzy logic is a convenient tool for expressing pieces of knowledge by the use of if-then rules. The FSRB consists of a block of fuzzy if-then rules whose consequences and antecedents are both fuzzy. The antecedents of a rule are fuzzy propositions describing the production environment such that "operation O_{ij} is urgent". The consequence of a rule is fuzzy, it can be expressed by "the priority of the operation O_{ijk} is high". In the FSRB, all propositions do not have the same degree of importance when contributing to the consequence.

For example, consider the rule with three propositions:

If "the job J_j is important" and "the operation O_{ij} is fast" and "the machine M_k is no-loaded"
then "the priority of O_{ijk} is high".

When a job is important and a no-loaded machine is available, we can give absolute priority to a slow operation of the job, since important job must be in time. These three propositions in the rule do not have the same weight in comparison with each other's.

The fuzzy rules, we consider in this paper are formalised as follow:

R_i: If p_1 and p_2 and ... p_n then c_i $\quad (\alpha_1, \alpha_2, \ldots, \alpha_n, \omega_1, \omega_2, \ldots, \omega_n, CF_i, \mu_i)$

Where:

1) $p_1, p_2, \ldots$, and p_n are scheduling propositions.
2) $\alpha_1, \alpha_2, \ldots, \alpha_n$ are respectively the degrees of belief of $p_1, p_2, \ldots$,and p_n , $\alpha_i \in [0, 1]$.
3) CF_i is a certitude factor associated to the rule R_i, $CF_i \in [0, 1]$.
4) c_i is the fuzzy consequence of the rule R_i.
5) μ_i is the membership degree of the consequence c_i, $\mu_i \in [0, 1]$.
6) $\omega_1, \omega_2, \ldots$, and ω_n are the membership degrees of $p_1, p_2, \ldots$,and p_n , $\omega_i \in [0, 1]$.

The parameters are defined such as the larger the value of α_k, the more the proposition p_k contributes to the decision c_i and the larger the value of CF_i the more we believe on the rule R_i.

The FSRB has the particularity that the consequent of a rule may be an antecedent for another rule. For example:

R_1: if "J_j is important" and "J_j is urgent" than "priority J_j is high"

R_2: if "priority J_j is high" and "O_{ij} is fast" than "priority O_{ij} is high"

The proposition "priority J_j is high" is a consequent of R1 and an antecedent of the rule R_2.

The fuzzy proposition conclusion part of the rules is represented by membership functions as described in figure1.

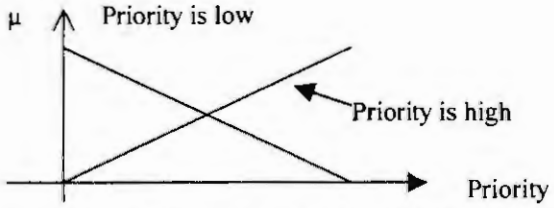


Figure 1: Consequent membership function

V. Interactive Fuzzy Scheduling Rule Base (IFSRB)

The increase of flexibility of production systems demands us to react to unpredictable events and configuration changes. Such configuration changes could be the introduction of a new important job to schedule, changes of the relative importance of the set of criteria or when adopting a new aggregated operator, or changing membership functions. The operator could be able to react to changes by modifying the **FSRB** parameters (degrees of truth and certitude factors) or by introducing and testing new scheduling rules.

In the case when the scheduler is dissatisfied with a ranking produced by the system, he can give his own point of view, his own decision. The mechanism must be able to perform a backward reasoning automatically and the operator is then asked to introduce his new scheduling rule if necessary or to change the certitude factor of some rules or the degrees of truth of some propositions. The scheduler is a simulator that solves chronologically the decision problems and reacts on the system in real time.

The mechanism works such that if configurations changes or the scheduler is dissatisfied with the decision given by the scheduling tool, the scheduler can test his own decision proposition and adopt the scheduling rule base to the state of the scheduling manufacturing environment.

Suppose that the operator is not satisfied with a decision given by the scheduling tool. He proposes to take into account his own point of view. The operator must then change the parameters of some rules (degree of belief) in order to obtain the convenient decision. In this case the decision given by the expert is called the goal decision c_j. A Backward Fuzzy Reasoning Algorithm (BFRA) is developed to find in the IFSRB the set of propositions that have as consequent the goal decision and asks the operator to adapt their degrees of belief values to his objectives. The set of propositions immediately preceding c_j is called PIP(c_j). Consider a fuzzy proposition p_i. If there is a rule in IFSRB in which p_i is in the antecedents propositions part and c_j is in the consequent part, we conclude that p_i is in PIP(c_j). The BFRA can be expressed by a fuzzy and/or graph. Each node of the tree is denoted by a quadruple (p_i, PIP(p_i), $\alpha(p_i)$, deg(p_i)), where:

p_i: proposition.

PIP(p_i): The set of propositions immediately preceding p_i.

$\alpha(p_i)$: degree of belief of p_i.

deg(p_i): degree of truth of p_i.

VI. Backward fuzzy reasoning algorithm

Input: goal decision c_j.

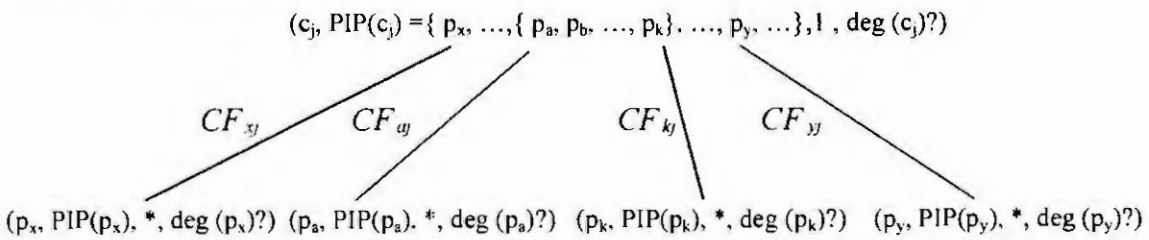
Output: modified interactive fuzzy scheduling rule base and the degree of truth deg(c_j) of the decision goal.

Step 1:

Initially, the root node is (c_j, PIP(c_j), $\alpha(c_j)$, deg (c_j)?). Since c_j is a consequent then The degree of belief $\alpha(c_j)$=1. deg (c_j)? is the degree of truth of the proposition c_j.

Step 2:

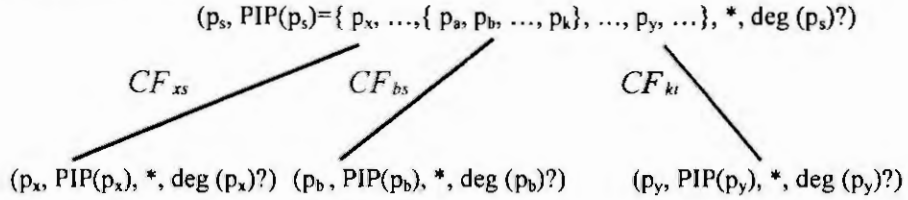
Sprout the fuzzy and/or graph as follow:



Where CF_{xj} is the certitude factor associated to the rule having c_j as consequent and p_x as antecedent.
In the quadruple (p_s, PIP(p_s), *, deg (p_s)?), the * means that the operator must give the value of the degree of belief associated to the proposition p_s and deg (p_s)? means that BFRA will evaluate automatically the degree of truth of p_s.

Step 3:

If PIP(p_s)=∅, then (p_s, PIP(p_s), *, deg (p_s)?) is called a terminal node.
If (p_s, PIP(p_s), *, deg (p_s)?) is a terminal node than select (p_s, PIP(p_s), *, deg (p_s)?)
Else sprout the fuzzy graph as follow:



Step 4:

If no terminal node exists,
go to step 3, else go to step 5.

Step 5:

Select a terminal node (p_t, ∅, *, deg (p_t)?) and ask the operator to introduce a value for the degree of belief $\alpha_t=\alpha(p_t)$. The degree of truth deg (p_t) of the terminal node is the membership degree of p_t. The terminal node becomes (p_t, ∅, α_t, deg (p_t)) and is said to be marked.

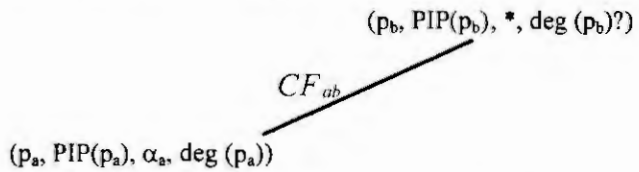
Step 6:

If all terminal nodes are marked go to step 7 else go to step 5.

Step 7:

Introduce the value $\alpha(p_k)$ of each non-terminal node (p_k, PIP(p_k), *, deg (p_k)?) and evaluate the degree of truth of p_k as follow:

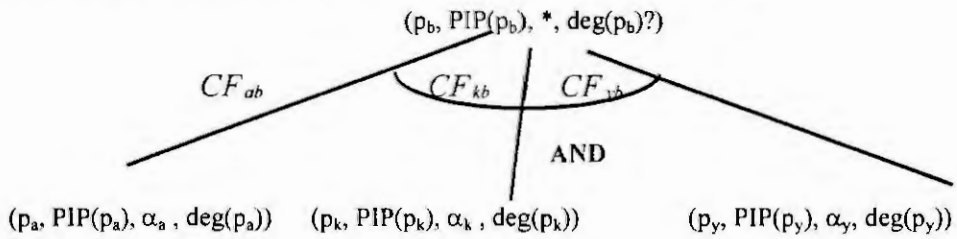
Case 1: if there is an edge of the graph shown as follow:



Then the node (p_b, PIP(p_b), *, deg (p_b)?) becomes (p_b, PIP(p_b), α_b, deg(p_b)) where:

α_b is introduced by the operator
deg(p_b) = α_a* deg (p_a) * CF_{ab} , deg(p_a) ∈[0,1] and CF_{ab} ∈[0,1]

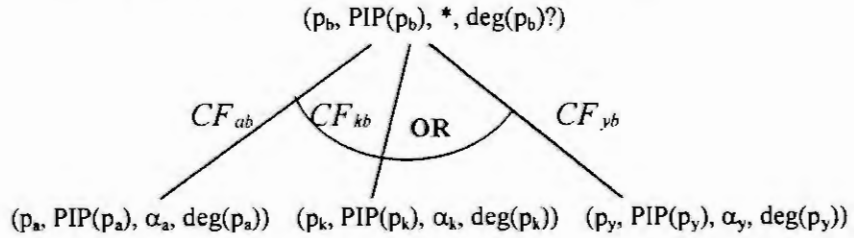
Case 2: If there exists an and subgraph in the generated graph shown as follow:



then the node (p_b, PIP(p_b), *, deg(p_b)?) becomes (p_b, PIP(p_b), α_b, deg(p_b)) where:

α_b is introduced by the operator
deg(p_b)= min[deg(p_a)* α_a* CF_{ab} , deg(p_k)* α_k* CF_{kb} , deg(p_y)* α_y* CF_{yb}] where
deg(p_a), deg(p_k) and deg(p_y) ∈[0,1]and CF_{ab}, CF_{kb} and CF_{yb} ∈[0,1]

case 3: If there exists an or subgraph in the generated graph shown as follow:



then the node (p_b. PIP(p_b), *, deg(p_b)?) becomes (p_b, PIP(p_b), α_b, deg(p_b)) where:

α_b is introduced by the operator

deg(p_b)= max[deg(p_a)* α_a* CF_{ab} , deg(p_k)* α_k* CF_{kb} , deg(p_y) α_y* CF_{yb}]

deg(p_a), deg(p_k) and deg(p_y) $\in$[0,1], CF_{ab}, CF_{kb} and CF_{yb} $\in$[0,1]

VII. Example

Let us have the rules:

R_1: If "O_{ij} is urgent" then "J_j is urgrent" (CF_1=0.95).
p_1 p_2

R_2: If "J_j is important" then "Priority J_j is high" (CF_2=0.9).
p_3 p_4

R_3: If "J_j is urgent" then "Priority J_j is high" (CF_3=0.8)
p_2 p_4

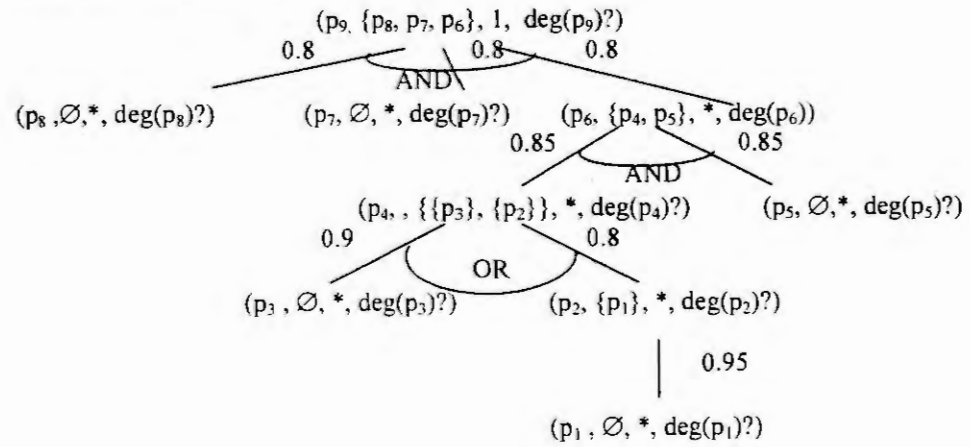
R_4: If "Priority J_j is high" and "O_{ij} is fast" then "priority O_{ij} is high" (CF_4=0.85).
P_4 p_5 p_6

R_5: If "M_k is no-loaded" and "priority O_{ij} is high" and "M_k is fast" then "priority O_{ijk} is high" (CF=0.8).
P_7 p_6 p_8 p_9

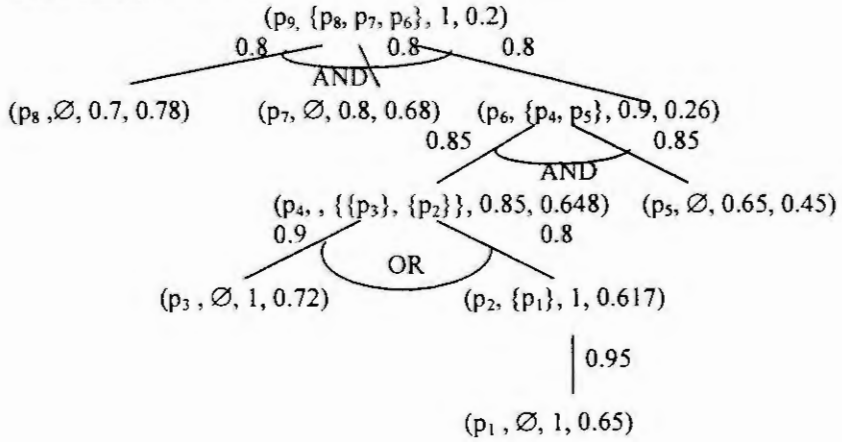
The scheduler wants to evaluate the degree of truth of the proposition p_9: "priority O_{ijk} is high". The first step consists of finding for each proposition p_i in the IFSRB the set of propositions immediately preceding p_i: PIP(p_i).

p_i	PIP(p_i)
p_1	∅
p_2	{p_1}
p_3	∅
p_4	{{p_3}, {p_2}}
p_5	∅
p_6	{p_4, p_5}
p_7	∅
p_8	∅
p_9	{p_8, p_7, p_6}

BFRA generates the following and/or graph:



The scheduler is asked to introduce the degree of belief value of each proposition in the and/or graph (α_4, α_5, α_6, α_8). The degree of truth of each terminal node is known. The input parameters of the system are as follow:
α_1=1, α_2=1, α_3=1, α_4=0.85, α_5=0.65, α_6=0.9, α_7=0.8, α_8=0.7, α_9=1.
deg(p_1)=0.65, deg(p_3)=0.72, deg(p_5)=0.45, deg(p_7)=0.68, deg(p_8)=0.78.
The BFRA will compute the values of respectively deg(p_2), deg(p_4), deg(p_6) and deg(p_9) by using Min-Prod and Max-Prod defuzzification methods. We obtain:



VIII. Conclusion

In this paper we have proposed a parametrable fuzzy scheduling rule base to deal with changes in scheduling environment. The scheduler is able to react to unpredictable events and configuration changes by adapting the values of the FSRB parameters. If the scheduler is dissatisfied with a decision given by the scheduling tool, he can evaluate the degree of truth of his own decision.

References:

1. B.Grabot, L.Geneste. Dispatching rules in scheduling: a fuzzy approach, INT.J.PROD.RES 1994, VOL. 32, NO. 4, 903-915.
2. C. Hershauer and J. Ebert, Search and simulation selection of a job-shop sequencing rule. Management science,1975, Vol. 21, No. 7.
3. Engell, S. et Moser M. A Survey of Priority Rules for FMS Scheduling and their performance for the Benchmark Problem, 1992, Proceedings 31st Conference I.E.E.E.
4. H. PIERREVAL , Expert system for selecting priority rules in flexible manufacturing systems. Expert. Sys. With Appl., Vol. 5, pp. 51-57.
5. MEBARKI, Une approche d'ordonnancement temps réel basée sur la sélection dynamique de règles de priorité, , Thèse de doctorat, Université Lyon1, Janvier 1995.
6. M.Montazeri and N.Van Wassenhove. Analysis of scheduling rules for an FMS, , INT.J.PROD.RES, 1990, VOL.28, NO. 4, 785-802.
7. N.Mebarki and H.Pierreval. Selection dynamique de règles de priorité pour les ateliers flexibles, 5th International Congress of Industrial Engineering., 1996, No. 2, p151-160.
8. N. LIOUANE, Contribution à l'élaboration d'un outil d'aide à la décision pour l'ordonnancement de production en environnement incertain, Thèse de doctorat , Université des Sciences et Technologies de Lille, Décembre 1998.
9. N.Louane, S.Hajri, S.Hammadi, P.Borne (fellow IEEE)
A robust fuzzy scheduling model for uncertain manufacturing systems
10. N.liouane, S.Hammadi, P.Borne, M.Annabi.
Piloting of Manufacturing system in uncertain environment. Efficient model of the uncertainty of the resources : hine, Information, IEEE, International Conference on SMC. Vol1,pp743-748 Orlando, Florida, USA 1997.
11. N.liouane, S.Hammadi, P.Borne, M.annabi, Robust methodologiy for scheduling production in uncertain environnnement, IMACS Multiconference CESA'98, Hammamet, Tunisie, 1998.
12. Panwalker, S.S. and Iskander. A survey of scheduling rules, Op. Research, 1977, Vol. 25, No. 1, pp. 45-61.
13. W. Conway, M. Johnson and L. Maxwell, An experimental investigation of priority dispatching. The journal of industrial engineering, 1960, Vol. 11, No. 3, pp. 221-229.

QUALITATIVE MODELING USING DYNAMIC FUZZY SYSTEMS

Klaus Schmid and Volker Krebs
Universität Karlsruhe (TH), Institut für Regelungs- und Steuerungssysteme
Kaiserstr. 12, D-76131 Karlsruhe, Germany
e-mail: {schmid, krebs}@irs.etec.uni-karlsruhe.de

Abstract. A dynamic fuzzy system is a mapping of fuzzy input values onto a fuzzy output value with a feedback to the input. In this paper, we present a new rule-based inference method that can be used in dynamic fuzzy systems. The inference result is always a fuzzy number. Therefore, the model output contains both, quantitative and qualitative information. Since this fuzzy output is fed back to the input, the dynamic fuzzy system models in particular the dynamic behavior of the qualitative information. A simulation example will demonstrate this feature of dynamic fuzzy systems.

1 Dynamic Fuzzy Systems

In most fuzzy systems an inference method maps the inputs onto the output according to an if-then rule base. The inference is a static mapping but it is possible to obtain a dynamic process model by embedding the inference in a dynamic structure. Then, the model output has to be fed back in order to represent the model dynamics. Two possible resulting model structures are shown in Figures 1 and 2.

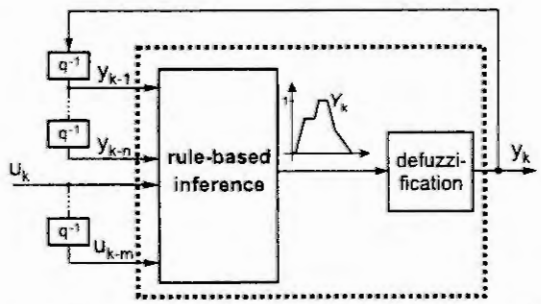


Fig. 1: Dynamic model with a static fuzzy system.

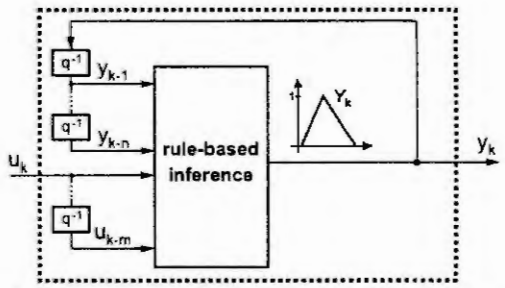


Fig. 2: Structure of a dynamic fuzzy system.

The dotted boxes enclose those parts processing fuzzy sets. Figure 1 represents the commonly used fuzzy approach to build a dynamic process model [4]. Note that in this case the feedback is not an element of the fuzzy system; the model consists of a *static* fuzzy system and the external dynamics. By contrast, the structure in Figure 2 feeds back fuzzy values and is therefore called *dynamic fuzzy system* [5]. The model output is a fuzzy set containing qualitative information about its accuracy. Since this output fuzzy set is used as an input variable in subsequent inference steps and since if-then rule bases usually are designed to process linguistic values, the membership function of the output fuzzy set must be interpretable as linguistic expressions like *almost zero* or *very large* [1, 5].

Therefore, in a dynamic fuzzy system we have to use a particular inference method fulfilling two basic requirements. First, the inference has to define a mapping of *fuzzy* inputs onto a *fuzzy* output. And second, all fuzzy sets (i.e. inputs, outputs, and linguistic values) must be represented by *interpretable* membership functions. Existing inference methods (e.g. the compositional rule of inference [2]) result in fuzzy sets with membership functions of a shape that is usually difficult to interpret. Hence, these inference methods are not suitable for dynamic fuzzy systems.

In section 2, we develop a new inference method for dynamic fuzzy systems. The features and the quality of dynamic fuzzy systems along with this inference method are demonstrated in a simulation example in section 3.

2 Inference with interpolating rules

Consider a simple example process with one input variable $\mathcal{E}$ and one output variable $\mathcal{Y}$. All we know about the process behavior is represented by a rule base consisting of two rules:

$$\begin{aligned} \mathrm{R}_1: &\quad \text{If} \quad \mathcal{E} = A \quad \text{Then} \quad \mathcal{Y} = K_A. \\ \mathrm{R}_2: &\quad \text{If} \quad \mathcal{E} = B \quad \text{Then} \quad \mathcal{Y} = K_B. \end{aligned}$$

The fuzzy sets A and B are called premises, K_A and K_B are called conclusions. All fuzzy sets have triangular membership functions that can easily be interpreted as linguistic values. The fuzzy sets are

depicted in Figure 3. When the current fuzzy input value E lies between the given premises (see Figure 3), the given rules cannot be evaluated directly. Yet it is clear that the output Y (which has to be computed by the inference method) should lie between the given conclusions. Furthermore, since the given input E is more fuzzy than the premises, it contains less information, and therefore the output should be more fuzzy than the conclusions, as well.

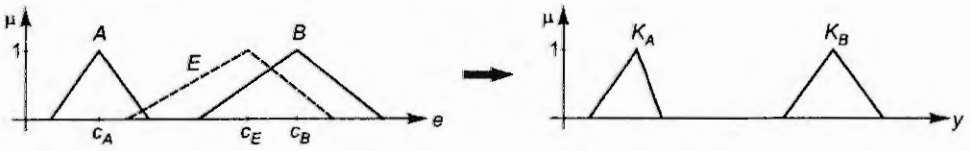


Fig. 3: Premises A and B, conclusions K_A and K_B, and input E.

In the following section we present the particular inference method required, the *inference with interpolating rules*. It consists of two steps: First, we generate an interpolating rule to determine the position of the output. After that, we map the fuzziness of the input onto the output.

2.1 Interpolating rule

The center of a triangular membership function (i.e. its peak value) determines the position of the fuzzy set on the input domain. The premise and the conclusion of the interpolating rule depend only on this center of the input fuzzy set, but not on the shape of its membership function [5]. Therefore, they can be calculated by interpolation with the center c_E of the input fuzzy set E as independent variable. The interpolating rule is given by the interpolating premise IP and the interpolating conclusion IC,

$$\text{IR: If } \mathcal{E} = IP \text{ Then } \mathcal{Y} = IC. \tag{1}$$

If c_E matches the center of one premise, the interpolating rule has to be equivalent to the corresponding given rule. Hence, the centers of the original premises serve as interpolation nodes with the given premises and conclusions as interpolation values. Using fuzzy sets as function values we can define the interpolation of triangular fuzzy sets. A triangular fuzzy set M is defined by three parameters, its center c_M, the left foot l_M, and the right foot r_M. Plotting these parameters versus the center of the input variable and applying linear interpolation we obtain the representation shown in Figure 4.

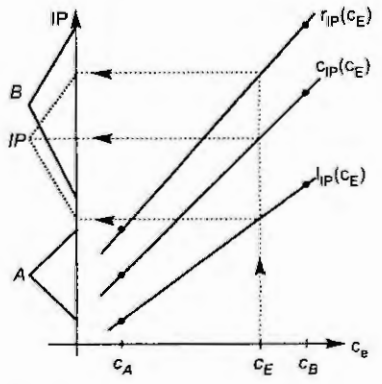


Fig. 4: Interpolation functions for IP, with the given premises A and B as interpolation nodes.

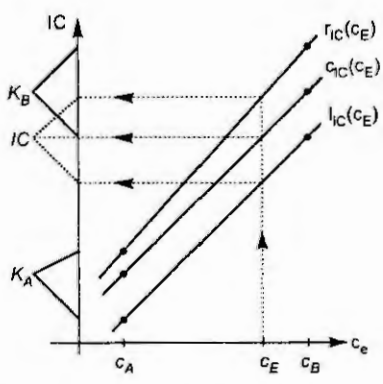


Fig. 5: Interpolation functions for IC, with the given conclusions K_A and K_B as interpolation nodes.

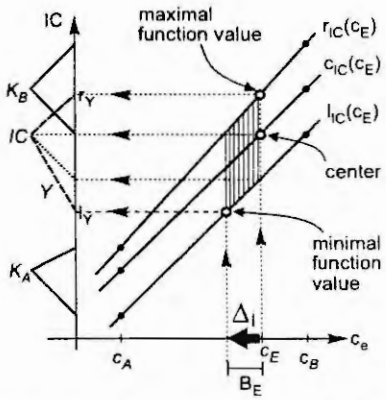


Fig. 6: Mapping the fuzziness: Maximal and minimal values are determined in the hatched area.

To obtain the triangular fuzzy set IP, three interpolation functions are necessary: $l_{IP}(c_E)$, $c_{IP}(c_E)$, and $r_{IP}(c_E)$ for the calculation of the left foot, the center, and the right foot, respectively. If now, for example, the input center c_E is equal to the center c_A of the given premise A, the interpolating premise is equivalent to premise A. If the center c_E lies between the centers of the premises A and B (as illustrated in Figure 4), the interpolating premise is calculated by the interpolation functions.

The interpolating conclusion IC is obtained in the same way, but then the conclusions K_A and K_B are used as interpolation nodes (see Figure 5). The interpolation functions need not be linear. They only have to be continuous functions and must not intersect in order to obtain interpretable fuzzy sets. Intersecting interpolation functions could, for example, result in a left foot of IP located right of c_{IP}.

2.2 Mapping the fuzziness

The interpolating conclusion determines the position of the output Y. In the second inference step we have to determine the fuzziness of Y. Since the rules describe the only knowledge about the behavior of the system, the output has to be at least as fuzzy as the interpolating conclusion. The interpolating premise and conclusion for the example mentioned above are depicted in Figure 7.

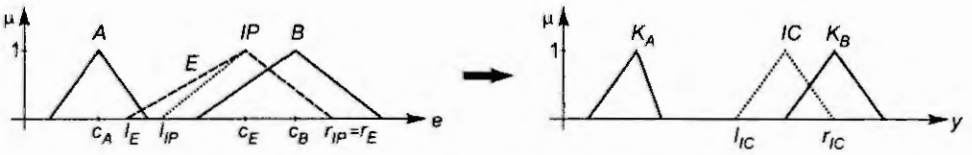


Fig. 7: Input E, interpolating premise IP, and interpolating conclusion IC.

In this example, the current input E is more fuzzy than the interpolating premise IP. We define the area below the fuzzy set's membership function as a measure of information content: the larger the area the smaller the information content. Applied to Figure 7, this means that E contains less information than IP. Therefore, it is quite clear that the output value Y should contain less information than the interpolating conclusion IC, as well. Hence, it follows that Y should be more fuzzy than IC.

As depicted in Figure 7, the left foot l_E of the input is more fuzzy than the left foot l_{IP} of the interpolating premise. When we consider membership functions as possibility distributions, this means that input values smaller than l_{IP} are possible. Therefore, for the given rule base, output values smaller than l_{IC} should be possible, as well. To map the fuzziness we compute the interpolation functions for IC not only at the single point c_E, but over a certain region B_E around that point. For the given example this region has to lie left of c_E, because the input E is more fuzzy than the interpolating premise on the left side. A possible B_E is shown in Figure 6.

The center of the output depends only on the center of the input and therefore is not changed by the mapping of the fuzziness; c_Y is always equal to the center of the interpolating conclusion [5]. The maximal possible value of Y in Figure 6 is represented by its right foot r_Y. Therefore, r_Y is determined as the maximal value of all three interpolation functions over the region B_E. The left foot l_Y is determined by the minimal function value. Applied to the input E of Figure 7, this mapping of fuzziness results in the output fuzzy set Y that is plotted in Figure 6 with dashed membership function.

One remaining degree of freedom is the absolute value to which we extend the considered input domain. Denoting the extension on the left and right side by Δ_l and Δ_r, we can define B_E in Figure 6 as $B_E = [c_E - \Delta_l, c_E + \Delta_r]$. For the given example, Δ_l is depicted in Figure 6. In this case, Δ_r is zero since the input fuzzy set's right side is not more fuzzy than the interpolating premise. A good choice for the amount of extension is the difference between the foot points of IP and the input fuzzy set:

$$\Delta_l = \begin{cases} l_{IP} - l_E & \text{for } l_E < l_{IP} \\ 0 & \text{otherwise} \end{cases}, \qquad \Delta_r = \begin{cases} r_E - r_{IP} & \text{for } r_E > r_{IP} \\ 0 & \text{otherwise} \end{cases}.$$

Whether an extension of the considered input domain to the left or right side affects the left or right foot of the output depends on the interpolation nodes. If the interpolation functions in Figure 6 had negative gradients, the extension Δ_l to the left side would affect the right foot of the output.

The extension of the presented inference method to systems with n input variables can simply be done by defining the interpolation functions over a multidimensional input domain. Then, an interval B_{Ei} is calculated separately for each input variable. Finally, the resulting domain B_E is determined as the Cartesian product of n intervals, $B_E = B_{E1} \times \cdots \times B_{En}$.

3 Simulation example

We use a very simple example to demonstrate the efficiency of dynamic fuzzy systems. For that purpose, we describe a first order dynamic system qualitatively by three rules

R_1 : If $y_{k-1} = 0$ And $u_{k-1} = 0$ Then $y_k = 0$.
R_2 : If $y_{k-1} = 1$ And $u_{k-1} = 0$ Then $y_k = 0.95$.
R_3 : If $y_{k-1} = 0$ And $u_{k-1} = 1$ Then $y_k = \langle 0.04/0.05/0.06 \rangle$.

The first rule represents the equilibrium point of the system, the third rule characterizes the influence of the input u on the system behavior. Since we do not know this influence precisely, we represent the conclusion of the third rule by a fuzzy set $\langle l/c/r \rangle = \langle 0.04/0.05/0.06 \rangle$. Note that the dynamic behavior

of this particular system could also be represented by a linear difference equation $y_k = ay_{k-1} + bu_{k-1}$ with an uncertain parameter b. The structure of the dynamic fuzzy system is depicted in Figure 8.

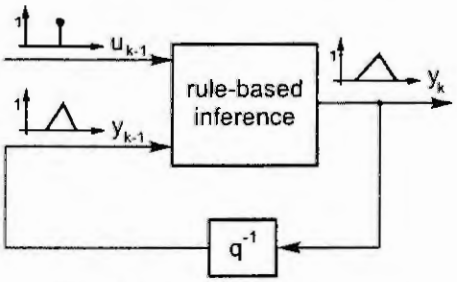


Fig. 8: Dynamic fuzzy system.

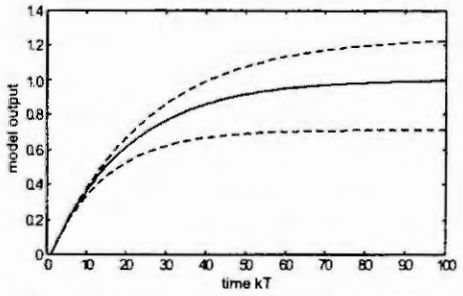


Fig. 9: Step response of the dynamic fuzzy system.

Since the fuzzy output is fed back to the input there is a loss of information in every time step. Starting with a crisp output value y_0 (and a constant crisp input $u = 1$), we obtain a triangular fuzzy set for the estimated output y_1. In the next time step, the dynamic fuzzy system computes an output y_2 that is even more fuzzy than y_1. Figure 9 illustrates the step response of this uncertain system. The model output at time kT is a triangular fuzzy set represented by its peak value (solid line), its left foot (lower dashed line), and its right foot (upper dashed line). This fuzzy set represents a region where the output of the modeled process can possibly lie. For example, the best estimate for the process output y_{100} is 1.0, but due to the uncertainty in our model the output could also lie between 0.7 and 1.2.

4 Summary

In this paper, we presented a new fuzzy inference method that – in contrast to existing fuzzy inferences – particularly considers the information content of inputs and outputs. The inference is based on the concept of interpolating rules, first calculating an interpolating rule, and in a second step mapping the fuzziness of the inputs onto the output. All fuzzy sets that occur can be interpreted as linguistic values. Therefore, this new inference method can be used in the feedback structure of dynamic fuzzy systems.

Dynamic fuzzy systems consider both, the qualitative and the quantitative informatiou contained in a linguistic representation of a dynamic process. Due to the transparency and the interpretability of this modeling approach it can also be used to identify a qualitative model for complex systems from data [3]. It is possible to model the time history of the information content that results from the uncertainties in the data or the linguistic values. Finally, since the inference is represented hy simple interpolation functions, further expert knowledge (e.g. concerning the equilibrium points of the system) can be used to improve the model quality [6].

References

[1] José Valente de Oliveira. A design methodology for fuzzy system interfaces. *IEEE Transactions on Fuzzy Systems*, 3(4):404–414, November 1995.

[2] Dimiter Driankov, Hans Hellendoorn, and Michael Reinfrank. *An Introduction to Fuzzy Control.* Springer-Verlag, Berlin, 1993.

[3] Volker Krebs and Elmar Schäfers. Dynamische Fuzzy-Systeme zur qualitativen Prozeßmodellierung. In *VDI-VDE/GMA-Tagung Computational Intelligence: neuronale Netze, evolutionäre Algorithmen, fuzzy control im industriellen Einsatz*, VDI-Berichte 1381, pages 115–135, Berlin, 1998.

[4] Oliver Nelles. *Nonlinear System Identification with Local Linear Neuro-Fuzzy Models.* Shaker Verlag, Aachen, 1999. Dissertation.

[5] Elmar Schäfers. *Dynamische Fuzzy-Systeme zur qualitativen Prozeßmodellierung: Eine neue Systemtheorie.* Fortschritt-Berichte VDI Reihe 8 Nr. 745. VDI Verlag, Düsseldorf, 1999. Dissertation.

[6] Klaus Schmid and Volker Krebs. Berücksichtigung von Vorwissen hei der Modellierung mit dynamischen Fuzzy-Systemen. In *Berichtsband zum 9. Workshop Fuzzy Control des GMA-FA 5.22*, pages 226–239, Dortmund, 1999.

MULTIPLE-CRITERIA DECISION MAKING USING τ -FUZZY MEASURE

N. Taira, H. Miyagi, K. Yamashita
Faculty of Engineering, University of Ryukyus
1 Senbaru, Nishihara, Okinawa 903-0213 Japan
E-mail: taira@sys.ie.u-ryukyu.ac.jp

Abstract. Fuzzy measure, proposed by Sugeno in 1972, is a mathematical concept. This concept does not assume additivity , it defines monotonicity and has been utilized in the originally proposed form. However, it is a very advanced mathematical concept compared to fuzzy set, so literature on developments in this area is limited. This report focuses on interaction between decision making factors, and proposes a new fuzzy measure.

1. Introduction.

As a measure of determining an ambiguity object, fuzzy measure[3] is well known and provides us with mathematical concepts on decision making problem. Probability measure, just like in physics measure such as the measurement of length or weight, are defined by positive semi definiteness, monotonicity and additivity. Fuzzy measure, on the other hand, is defined by positive semi definiteness and monotonicity thus subsuming probability measure as a special type.

The definition of fuzzy measure provides us with conditions of set function. However, because of loose restrictions compared to probability measure, fuzzy measure makes a problem of determining values of these on all subsets of X , which are difficult to obtain, even if X is a finite set.

Decomposable measure[5], proposed by Weber, provides us with conditions for dealing with the problem. λ -fuzzy measure is a set function satisfying conditions as decomposable measure and has been widely utilized in decision making problem. It is not natural, however, to consider that λ -fuzzy measure can express all human subjective images. It is natural to consider that the images can be expressed by some set functions. Thus, it is debatable researching set functions which reflect the images.

This report focuses on an interaction between decision making factors which is needed to determine an estimation, and proposes τ -fuzzy measure as a new fuzzy measure. This paper is arranged as follows: Section 2 provides definitions of the important terminologies used. In section 3, we propose the use of meibus inversion, deal with interaction of λ -fuzzy measure, and consider a way of expressing interaction using an example. Moreover, τ -fuzzy measure is proposed as a new fuzzy measure. Section 4 gives a selecting problem of suitable power station system as an example, applying τ -fuzzy measure. Finally, we conclude this report in section5.

2. Definitions.

In this section, we provide definitions used in the report. Let X be a non-empty set and let F be a σ -algebra defined on X.

Definition 1 (Fuzzy Measure) [3]

A fuzzy measure on a measurable space(X,F) is a set function $\mu : F \rightarrow [0,\infty)$ with following axioms:

Axiom 1 (positive semi definiteness) : $\mu(\phi) = 0, 0 \leqq \mu(A) \leqq \infty$,

Axiom 2 (monotonicity) : A, B$\in F$ and if A$\subseteq$B, then $\mu(A) \leqq \mu(B)$,

Axiom 3 (continuity) : For every increasing (or decreasing) sequence $\{A_i\}$ of F

$$\lim_{i\to\infty} \mu(A_i) = \mu(\lim_{i\to\infty} A_i).$$

Definition 2 (λ -Fuzzy Measure) [1]

A fuzzy measure μ_λ is called λ-fuzzy measure (or Sugeno measure) if it satisfies, in addition to axioms 1 through 3 of fuzzy measure, the following special axiom: For all $A,B \in F$, if $A \cap B = \phi$, then

$$\mu_\lambda(A \cup B) = \mu_\lambda(A) + \mu_\lambda(B) + \lambda\, \mu_\lambda(A)\, \mu_\lambda(B), \tag{1}$$

$$-1 < \lambda < \infty . \tag{2}$$

Where the inequality (2) is the condition of equation (1) to satisfy axiom 2. The general formula of equation (1) is

$$\mu_\lambda\left(\sum_{i=1}^{\infty} A_i\right) = \frac{1}{\lambda}\left[\prod_{i=1}^{\infty}(1+\lambda\mu_\lambda(A_i)) - 1\right] . \tag{3}$$

The λ-fuzzy measure can be considered to be belief measure for $\lambda > 0$, and to be plausibility measure for $\lambda < 0$, and to be probability measure for $\lambda = 0$.

Mebius inversion is the very important theorem on integral theory, it guarantees the existence of inversion function for a defined function on integer system. Definition of mebius inversion applying set function is as follows;

Definition 3 (Mebius Inversion) [2]

When X is finite set, set function $\mu, \nu : 2^X \rightarrow \mathbf{R}$ can be associated by

$$\mu(A) = \sum_{B \subseteq A} \nu(B), \forall A \in 2^X . \tag{4}$$

This correspondence proves to be one-to-one, since conversely

$$\nu(A) = \sum_{B \subseteq A} M(B,A) \cdot \mu(B), M(B,A) = (-1)^{|A-B|} \text{ for } B \subseteq A \subseteq X. \tag{5}$$

Here, M(B,A) is called mebius function, ν is called mebius inversion of μ.

Definition 4 (Choquet integral) [3]

The Choquet integral of a measurable function $f : X \rightarrow \mathbf{R}$ with respect to a fuzzy measure μ is defined by

$$(C)\int_A f\, d\mu = \int_{-\infty}^{+\infty} \mu_f(r)\, dr \tag{6}$$

where

$$\mu_f(r) = \begin{cases} \mu(\{x \mid f(x) > r\} \cap A) & \text{if } r > 0 , \\ \mu(\{x \mid f(x) > r\} \cap A) - \mu(A) & \text{if } r < 0 . \end{cases} \tag{7}$$

3. τ -Fuzzy measure.

Considering Definition 3, fuzzy measure μ (A) can be represented by mebius inversions, which are composed of power sets of A, mebius inversions of μ (A) can be regarded as set functions expressing the interactions between factors of A. On the one hand, calculating mebius inversion for λ-fuzzy measure, the result is the following equation,

$$\nu(A) = \lambda^{|A|-1} \prod_{a \in A} \mu(a) . \tag{8}$$

This equation indicates that the interactions between factors of A are influenced by μ (a). However, this relation does not fit our image in some situations. For instance, the following labor's estimate problem can be given;

Worker A has experience and worker B has no experience. Then, we can expect that the anticipation of the manager is μ (worker A)>0, μ (worker B)=0 in case of working separately. In contrast, if the workers work together, then we can expect that the anticipation of the manager is μ (worker A $\cup$ worker B)> μ (worker A) (or μ (worker A $\cup$ worker B)< μ (worker A)).

In the above case, applying λ -fuzzy measure, μ (worker A $\cup$ worker B) must be equal to μ (worker A) and equation (8) leads to the result that the interaction of worker A and worker B must be zero.

Therefore, this report considers the interaction as a value of potential possibility and defines

$$v(A) = \sum_{k=1}^{|A|} \tau_k \tag{9}$$

as the interaction. We define τ -fuzzy measure leading to the set function applying mebius inversion for the equation(9).

Definition (τ -Fuzzy Measure)

A fuzzy measure μ_τ is called τ -fuzzy measure if it satisfies, in addition to axioms 1 through 3 of fuzzy measures, the following special axiom: for all $A_1, A_2 \in F$, $\mu_\tau(A_1) \geqq \mu_\tau(A_2)$,and if $A_1 \cap A_2 = \phi$, then

$$\mu_\tau(A_1 \cup A_2) = \mu_\tau(A_1) + \mu_\tau(A_2) + \tau_1 + \tau_2, \tag{10}$$

$$\tau_1 + \tau_2 > -\mu_\tau(A_2). \tag{11}$$

Where the inequality (11) is the condition of equation (9) to satisfy axiom 2. The general formula of equation (10) and (11), for every $\mu_\tau(A_i) \geqq \mu_\tau(A_j)$ ($i < j$) and $A_i \cap A_j = \phi$ ($i \neq j$), is

$$\mu_\tau\left(\sum_{A_i \in A} A_i\right) = \sum_{A_i \in A} \mu_\tau(A_i) + (2^{|A|-1} - 1) \sum_{k=1}^{|A|} \tau_k \ , \tag{12}$$

$$\tau_i \geq \frac{1}{2^{i-1}-1}\left[\frac{2^{i-2}}{2^{i-2}-1} \sum_{k=2}^{i-1} \mu_\tau(A_k)\right] . \tag{13}$$

Applying τ -fuzzy measure for the labor's estimate problem, we can get more closer estimate of our image on the ground that the restriction of this measure's interaction is more loose compared to other measures. Furthermore, τ -fuzzy measures can be regarded as probability measures for every $\tau = 0$, and as λ -fuzzy measure for

$$\tau_i = -\frac{1}{2^{i-1}-1} \mu(A_i)\left[\prod_{k=1}^{i-1}(1 + \lambda\mu(A_k)) - 1\right] \qquad (\because 1 \leq i \leq |A|,\ \tau_1 = 0) . \tag{14}$$

4. Simulation.

In this section, we deal with the selecting problem of suitable power station systems as an example, and confirm that τ -fuzzy measure is suitable for expressing subjectivity judgment. If some kinds of alternative systems, which could be selected, are given to a decision maker(DM), then the DM could determine these suitability, such as values, under his knowledge or images. The answers could be treated as estimations including his ambiguity. Assuming there are four kinds of systems (thermal power station, wind power station, solar power station, and nuclear power station) which could be selected as a suitable system, and that we have the following answer of DM which are considered as suitability of these systems;

(thermal system, wind system, solar system, nuclear system)=(0.23, 0.27, 0.31, 0.19)

and comprehend that these are determined under assessment bases which are safety, scene, stability supply and electric charge.

Evaluating our measure, this report handles Choquet integral using this measure as a model of the above problem, and confirms how close it could come to the given estimation. Choquet integral is the extension of Lubesgue integral and is a concept for dealing with fuzzy measure. Here, fuzzy measure is used in measuring the assessment bases which gives suitability of these systems. Moreover, it shows that Lubesgue integral model which does not use fuzzy measure and Choquet integral model using λ -fuzzy measure could come close to given estimation.

On processing of Choquet integral, however, weights of decision making ingredients and values of invariable τ are needed. Here we have determined these weights using Analytic Hierarchy Process(AHP)[6] and the values of τ as minimizing the square average error between calculated estimations and given estimations. Here, AHP,

proposed by T.L.Saaty, is a way of scoring human subjectivity and is well known as an effective technique[4]. The square average error is given as equation (15).

$$J = \sqrt{\frac{1}{4}\sum_{i=1}^{4}(d_i - e_i)^2} \quad \text{d: given estimation, e: calculated estimation} \tag{15}$$

The results of this simulation are shown in the table below.

TABLE The results of simulation

	calculated estimations				invariable using fuzzy measure	square average error (J)
	thermal system	wind system	solar system	nuclear system		
Choquet integral model (using τ -fuzzy measure)	0.2300	0.2700	0.3084	0.1900	$\tau_{safety} = -0.1354$ $\tau_{scene} = 0.0544$ $\tau_{stability\ supply} = 0.2259$ $\tau_{electric\ charge} = -0.0430$	0.0006
Choquet integral model (using λ -fuzzy measure)	0.1541	0.2349	0.4735	0.1374	$\lambda = -0.656$	0.0949
Lubesgue integral model (no using fuzzy measure)	0.1575	0.2312	0.4758	0.1338		0.0974

The above results show that Choquet integral model could determine, compared to Lubesgue integral model, closer to human judgments. And, it could be confirmed that τ -fuzzy measure is suitable for expressing ambiguous judgments and, in this simulation, determines closer to human judgments compared to λ -fuzzy measure. The reason for less accurate result in using λ -fuzzy measure could be understood to be the result of serious restrictions in the interactions λ -fuzzy measure.

5. Conclusions.

In this report, we focused on the interaction of human subjectivity and considered how to deal with interaction of λ -fuzzy measure using mebius inversion formula. The τ -fuzzy measure proposed in this report has less restrictions compared to λ -fuzzy measure and can estimate more closer to human subjectivity. Future work will deal with the determination algorithm of the invariable τ .

Acknowledgements. The authors gratefully acknowledge the financial assistance from SCAT.

6. References.

1 George J. Klir and Tina A. Folger, Fuzzy sets, uncertainty, and information. Prentice Hall, 1988.

2 K. Fujimoto, Decision Making and Mebius Inversion. In: Journal of Japan Society for Fuzzy Theory and Systems, Vol.10, No.2, 1998, 206-214.

3 M. Sugeno, Fuzzy Measure and Fuzzy Integral. In: Transactions of the Society of Instrument and Control Engineers, Vol.8, No.2, 1972, 218-226.

4 H. Miyagi, N. Taira and K. Yamashita, Multiple-Criteria Decision Making Using Isomorphism. In: Proceedings of IEEE International Conference on SMC, Vancouver, Vol.1, 1995, 748-752.

5 S. Weber, $\perp$ -decomposable measures and integrals for Archimedean t-conorms $\perp$, In: J. Math. Anal., Vol.101, 1984, 197-222.

6 T. L Saaty, The Analytic Hierarchy Process. McGraw-Hill, 1980.

MODELING A HYDRAULIC DRIVE USING NEURAL NETWORKS

Carsten Otto

Department of Measurement and Control (Prof. Dr. H. Schwarz)
University of Duisburg, 47048 Duisburg, Germany
e-mail: co@uni-duisburg.de

Abstract This paper presents the nonlinear black box modeling of a hydraulic translatory drive using neural networks. The type of neural network employed here is the multilayer perceptron. Feeding previous inputs and outputs into the network leads to two different black box model structures, namely the series-parallel and the parallel model. Their suitability for modeling the hydraulic drive on the basis of measurements on a test bed is compared.

1 Introduction

Modeling of technical systems using physical laws can be quite difficult if there is not enough physical insight to the system or if the resulting mathematical equations become too complex. If only the input-output behaviour of the system is of interest and knowledge is gathered from experimental data, it is useful to choose a black box approach to obtain a model of the technical system [8]. In recent years artificial neural networks have gained importance in nonlinear black box modeling [1], [6], [7].

The multilayer perceptron, which is one of the most popular types of artificial neural networks, is known to be a universal approximator of nonlinear relationships [3]. In order to model the dynamic behaviour of the underlying system, historical information, i. e. past inputs and outputs, has to be used as input into the network. In this context two different nonlinear black box model structures can be considered: the series-parallel model, using past inputs and *measured* outputs $y(k-j)$, and the parallel model, using past inputs and *predicted* outputs $\hat{y}(k-j)$. The objective of this paper is to discuss these two black box model structures for modeling a hydraulic translatory drive with the multilayer perceptron.

Section 2 describes the multilayer perceptron used in this paper and in section 3 the two black box model structures will be introduced. Section 4 illustrates the hydraulic translatory drive and section 5 presents the results of modeling the drive. The paper closes with a summary.

2 Neural networks

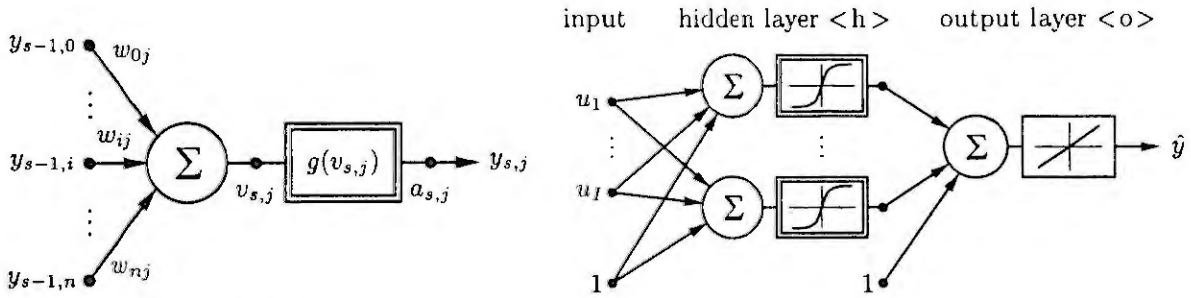
Neural networks can be described as signal processing systems made up of simple units, which communicate through weighted connections. The multilayer perceptron (MLP) is one of the most popular types of neural networks. Here, the units are arranged into one or more hidden layers and an output layer. Units within successive layers are coupled by weighted connectious. The input signals propagate through the network in a forward direction [2].

Fig. 1(a) depicts the structure of a unit in layer s. It is fed by all output signals $y_{s-1,i}$ from the n units of the preceding layer and calculates the activation potential

$$v_{s,j} = w_{0j} y_{s-1,0} + \sum_{i=1}^{n} y_{s-1,i} w_{ij} \ , \tag{1}$$

using the weights w_{ij}. The activation function $g(v_{s,j})$ calculates the activation $a_{s,j}$, which is sent as an output $y_{s,j}$ to the units of the subsequent layer. A bias is applied to the unit, represented by a constant signal $y_{s-1,0} = 1$ and its weight w_{0j}.

This paper deals with MLPs consisting of one hidden layer and a single unit in the output layer (fig. 1(b)). The hyperbolic tangent function is used as the activation function $g(v_{s,j}) = \tanh v_{s,j}$ for



(a) Unit (b) Network

Figure 1: Multilayer Perceptron (MLP)

the units in the hidden layer, while the identity function is applied to the output unit. The functional relationship between the output $\hat{y}$ and the input $\boldsymbol{u} = [u_1, \ldots, u_I]$ is given by

$$\hat{y}(\boldsymbol{u}, \boldsymbol{w}) = w_{01}^{<o>} + \sum_{j=1}^{H} w_{j1}^{<o>} \tanh\left(w_{0j}^{<h>} + \sum_{i=1}^{I} w_{ij}^{<h>} u_i\right) , \tag{2}$$

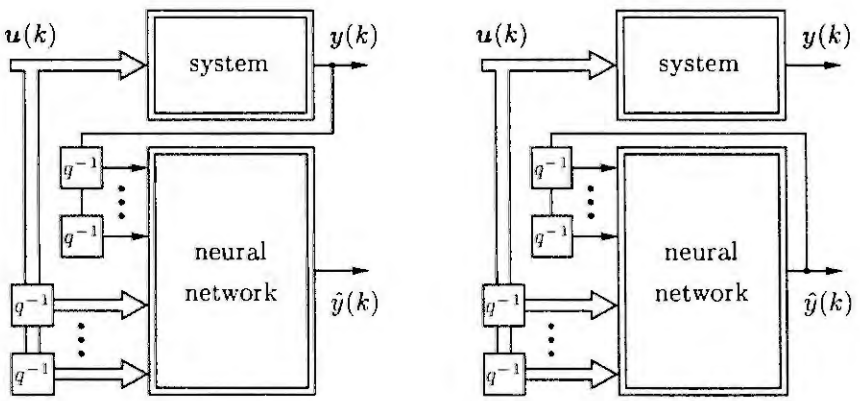
where I denotes the number of input signals and H the number of hidden units. During a training phase the MLP learns the functional relationship (2) on the basis of N measured data points. The aim of learning algorithms is to minimize some error criteria, for example the average squared error

$$V = \frac{1}{N} \sum_{p=1}^{N} \left(y_p - \hat{y}_p\right)^2 . \tag{3}$$

In this contribution, minimization of (3) is achieved using the Levenberg-Marquardt method [7].

3 Black box models

In order to model the dynamic behaviour of technical processes, historical information, i. e. past inputs and past outputs, has to be used as input to the MLP. The resulting dynamic behaviour of the MLP is essentially influenced by the way the output is fed back into the network. Two approaches often discussed in the literature are the series-parallel and the parallel model [5], also named NARX and NOE models



(a) Series-parallel model (b) Parallel model

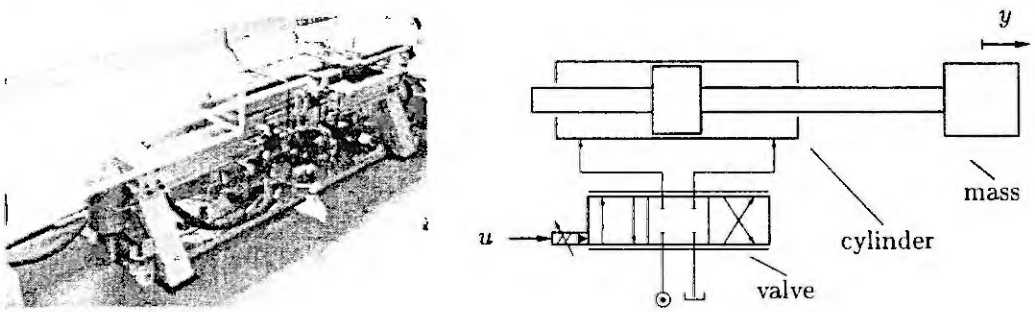
Figure 2: Black box models

respectively [9]. In the case of the parallel model in fig. 2(b) the predicted output is fed back as input to the model, i. e. there is complete parallelism between model and system, while parallelism for the series-parallel model in fig. 2(a) is only given regarding the input.

Both approaches can be used for identification of a model as well as for validation. The calculation of the gradients for optimization of (3) is far more complex for identification of a parallel model, since the output depends on previous outputs. The identification of a series-parallel model results generally in higher accuracy. However, regarding the validation aspect it can be more suitable to identify a parallel model as will be shown in section 5.

4 Description of the hydraulic drive

The hydraulic translatory drive depicted in fig. 3 consists of a synchronized cylinder and a servo valve, that controls the oil pressure inside the two cylinder chambers and is actuated by the voltage u. The task is to predict the velocity v of the piston depending on u. Identification data is gathered by actuating



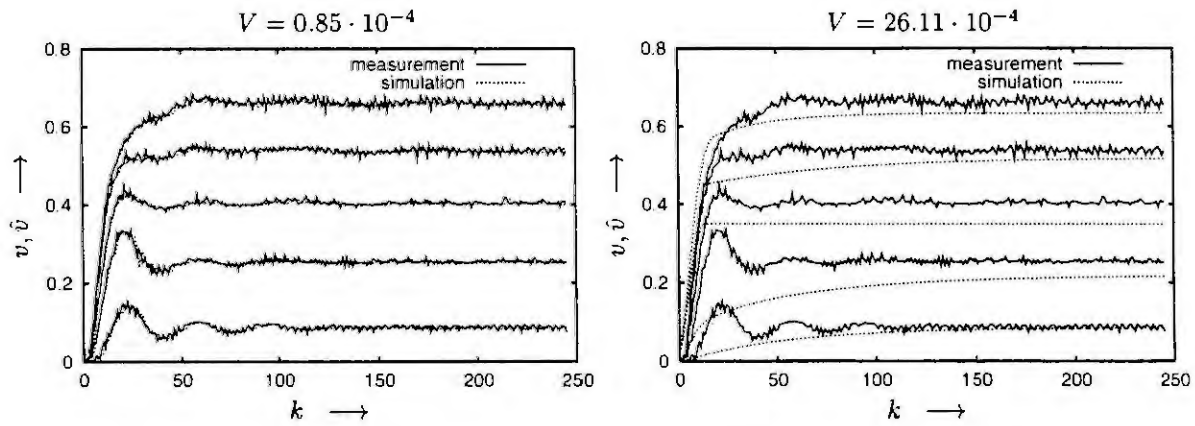
(a) Test bed (b) Schematic diagram

Figure 3: Hydraulic drive

the drive with an amplitude modulated pseudo random signal. Data for validation consists of several step responses. For black box modeling, the input $u(k-4)$ and the outputs $v(k-1),\ldots,v(k-4)$ and $\hat{v}(k-1),\ldots,\hat{v}(k-4)$ respectively are used as regressors. For further details on the drive and regressor structure see [4].

5 Modeling results

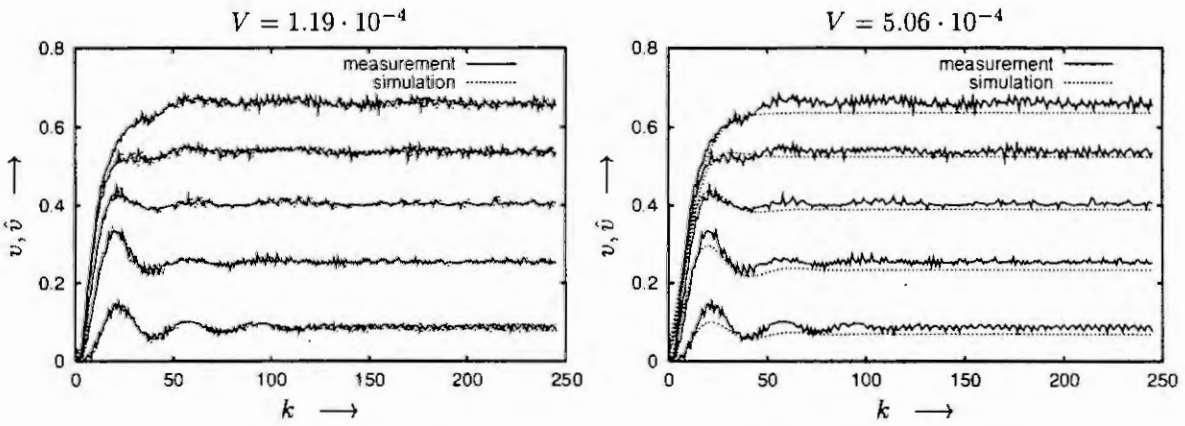
This section presents the results of modeling the hydraulic drive with a MLP using 6 units in the hidden layer turning out a model having 43 parameters (weights of the MLP). The identification is done as a series-parallel model as well as a parallel model. Each of the two resulting models is then validated in series-parallel mode and in parallel mode in order to compare them.



(a) Series-parallel mode validation (b) Parallel mode validation

Figure 4: Results for the series-parallel model

The validation results for modeling the drive in series-parallel mode are shown in fig. 4 and the validation results for modeling in parallel mode are shown in fig. 5. Comparing the case the model is validated in the same mode for which it was identified (fig. 4(a) for the series-parallel model and fig. 5(b) for the parallel model), the series-parallel structure leads to higher accuracy. But comparing the case the models are validated using the opposite mode, the series-parallel model (fig. 4(b)) is unstable, whereas the parallel model (fig. 5(a)) still yields satisfactory results.



(a) Series-parallel mode validation

(b) Parallel mode validation

Figure 5: Results for the parallel model

6 Summary

Neural networks provide a good approach for nonlinear black box modeling of technical processes, as is shown in this paper for a hydraulic translatory drive. However, the user has to decide which kind of black box structure to apply. As the results in this contribution show, if the measurement of the system's output is not desired and the model is running in parallel mode, it is more suitable in practice to identify the model in parallel mode as well, although the identification procedure becomes more complex.

References

[1] Mihiar Ayoubi. *Nonlinear System Identification Based on Neural Networks with Locally Distributed Dynamics and Applications to Technical Processes.* Fortschritt-Berichte VDI Reihe 8 Nr. 591. VDI, Düsseldorf, 1996.

[2] Simon Haykin. *Neural Networks.* Prentice-Hall, New Jersey, 1999.

[3] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2:359–366, 1989.

[4] Andreas Kroll and Angelika Agte. Modellierung hydraulischer Antriebe durch Fuzzy-Modelle. *Ölhydraulik und Pneumatik*, 41(6):414–416, 1997.

[5] Kumpati S. Narendra and Kannan Parthasarathy. Identification and control of dynamical systems using neural networks. *IEEE Trans. on Neural Networks*, 1(1):4–27, 1990.

[6] Oliver Nelles. *Nonlinear System Identification with Local Linear Neuro-Fuzzy Models.* Berichte aus der Automatisierungstechnik. Shaker, Aachen, 1999.

[7] Magnus Nørgaard. *System Identification and Control with Neural Networks.* Ph.d. thesis, Department of Automation, Technical University of Denmark, Lyngby, Denmark, 1996.

[8] Helmut Schwarz. *Einführung in die Systemtheorie nichtlinearer Regelungen.* Shaker, Aachen, 1999.

[9] Jonas Sjöberg et al. Nonlinear black-box modeling in system identification: a unified overview. *Automatica*, 31(12):1691–1724, 1995.

Homogenous Neural Network prepared for Interferometry Images.

Z.Gomólka
Rzeszów Pedagogical University. Institute of Technology
Rzeszów 55-059, 16A Rejtana str, Poland
e-mail: zgomolka@atena.univ.rzeszow.pl

Abstract:
Paper deals with new structures of neural network designed for detection of maxims and ridges in the two dimensional signals especially in interferometry images. These images called sometimes fringe images have been obtained by Moire projector. They have a wide variety of applications in the non touch measurement techniques applied for shape, stress, displacement analysis and many others. The known architectures of the investigated network for one dimensional performance have been extended to the two dimensional case. The different types of architectures which realize this task have been discussed. The influence of the two crucial parameters of neural network ie. its value of the weights in the input layer and the number of neurones establish sensitivity of the network to noise suppression. The most efficient hexagonal architecture have been compared to classical skeletoning algorithms. The proposed net can provide the same results as other known methods but in shorter time according to the parallel performance of the net.

1. Generation of interferometry images

The classical device for Moiré images is presented below on the Fig.1. From the left side we have the source of light which passes through the standard optical line $E'_2 - E_2$ comprised with special grid G. This grid produces set of fringes on the surface of the illuminated object. From the right side camera CCD watch these fringes through the another optical line $E'_1 - E_1$. In case the surface of the object is ideally flat these fringes are parallel to each other. Otherwise they have different phase and amplitude according to the amount of the surface distortion. By sliding the intersections of the obtained image we could get the intensity distribution as on the Fig.2. The function of intensity of such image called also the wave equation might be expressed as:

$$I(x,y) = a(x,y) + b(x,y)\cos[\varphi(x,y)]$$

where:

- $I(x,y)$ denotes the intensity value at point (x,y),
- $a(x,y)$ denotes background intensity at this point,
- $b(x,y)$ contrast of fringes,
- $\varphi(x,y)$ desired phase function.

Distribution of phase or location of maxims encompasses crucial information about the shape of specimen or it's distortion. In this paper we consider the second circumvention which in general is belonged the intensity methods. The noise caused by optical system states inseparable component (see Fig.2) which have to be removed during detecting. It is well known that usually the edges or ridges in the image cover the wide spectrum and this point causes essential problem in detecting process. Historically first time the necessity of the image decomposition to the complementary frequencies have been voted by Marr

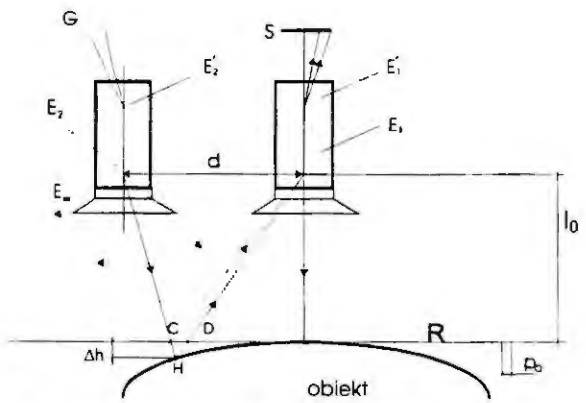


Fig. 1 View of the optical system used in the experiments.

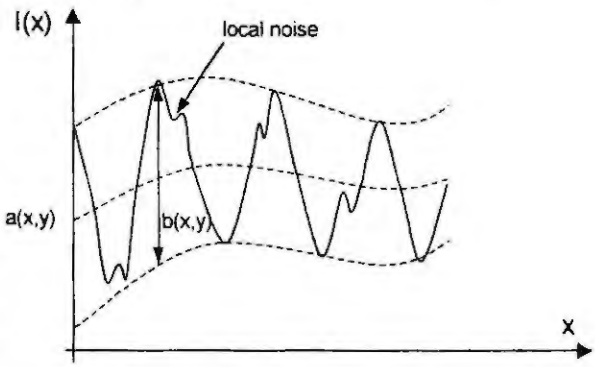


Fig. 2 Intensity distribution across interferometry image.

and Hildreth. Their idea extended Canny and eventually Mallat proposed ***MRA*** model [Edw91[1]], [Vid94[2]]. The *state of art.* might be found in [Sta96[3]],[Hsi97[4]] and [Hea98[5]]. Proposed in the next chapter network in some details refers to this model, and will state this base model for the further investigation with hierarchical neural network.

2. Neural network for maxims detection

Architecture of the network for maxim indication have been presented in details in earlier works [Dud Gom96[6]], [Gom99[7]]. Very interesting point is the model of the neural network architecture very like biological visual system of the crab *limulus*. Another resemblance might be indicated to the performance of the human ear system. A brief recall of the assumptions we will present here. Network consist of the two layers with homogenous weight distribution.

$$w_r(i)=\begin{cases}(-1)^{r-i}\cdot\binom{r}{i} & for \quad i\in[0,r]\\ 0 & for \quad i\notin[0,r]\end{cases} \tag{1}$$

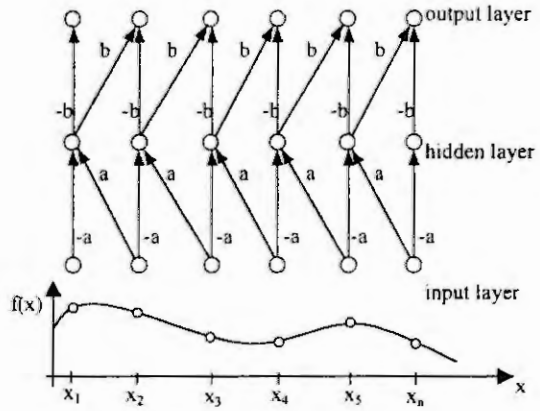


Fig.3. The network architecture.

where i denotes neuron number, and r denotes number of the measured difference. Thus the input signal to the neuron n will become:

$$e_r(n)=\sum_{i=-\infty}^{+\infty} w_r(i)\cdot f_{r-1}(n+i)=\Delta^r f_{r-1}(n) \tag{2}$$

So we might say , the network analyses the discrete input functions $f_0(n) = f(x_n)$. The transfer function $h(n)$ of the predetermined neuron in the first layer has the form:

$$h(e)=\begin{cases}0 & for \quad e<-p\\ k(e+p)/2p & for \quad -p\le e\le p\\ k & for \quad e>p\end{cases} \tag{3}$$

where p and k are parameters of positive values.

Assuming homogenous weighting of the signal in the first layer by value a and by b in the second layer we have:

$$e_1(n)=\sum_{i=-q}^{q} w_1(i)\cdot f_0(n+i)=-af_0(n)+af_0(n+1)=a\Delta^1 f_0(n) \tag{4}$$

and:

$$e_2(n)=\sum_{i=-q}^{q} w_2(i)\cdot f_1(n+i)=bf_1(n-1)-bf_1(n)=-b\Delta^2 f_1(n-1) \tag{5}$$

These equations might be called the neuron like version of the convolution in the signal analysis theory. It is possible to prove that points n which fulfill two above conditions states maxims in the input signal: (it is neural realization of *zero-crossing* problem*)*:

$$\Delta^1 f_0(n-1)>0 \quad \wedge \quad \Delta^1 f_0(n)<0 \tag{6}$$

The assumed transfer function for the both layers are depicted on the Fig.4.

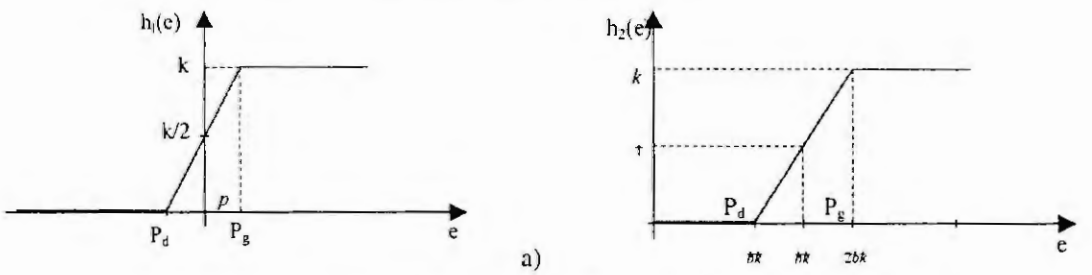


Fig.4. The transfer functions in the first layer a), and the second layer b).

where the transfer of the second layer have been stated as:

$$f_2(n)=h_2(e(n))=\begin{cases}0 & for \quad -b\Delta^2 f_1(n-1)<p_d\\ \dfrac{-b\Delta^2 f_1(n-1)-p_d}{(p_g-p_d)} & for \quad p_d\le -b\Delta^2 f_1(n-1)\le p_g\\ 1 & for \quad -b\Delta^2 f_1(n-1)>p_g\end{cases} \tag{7}$$

By using assumed notation we can extend the condition for the "maxims presence" in the second layer as:

$$-\Delta^2 f_1(n-1) > \frac{k}{2} \tag{8}$$

The flow of the signal through the network presents another figure:

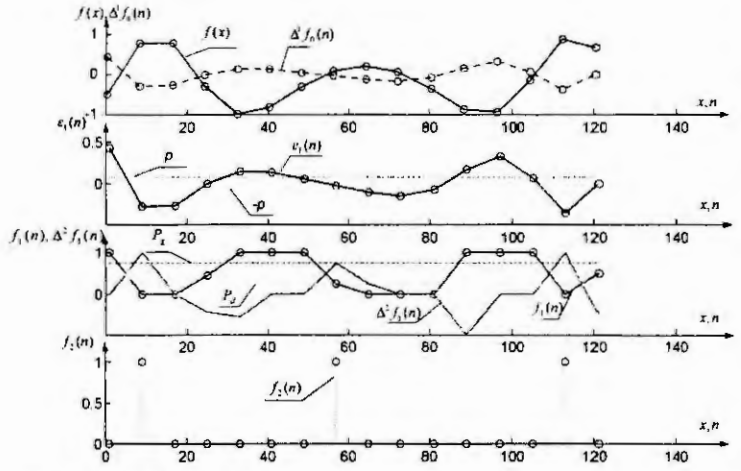


Fig.5. The flow of the signal through the network.

From the top of the figure we have input signal $f(x)$, and the graph of the first difference $\Delta^1 f_0(n)$. The value of the a parameter has crucial influence on the shape of the $e_1(n)$ signal. Second graph presents threshold process at the first layer. Signal $\Delta^2 f_1(n)$ might be called the extracted *zero-crossings* with the same slope. At the bottom we have output signal from the network . The medial part of the $\Delta^2 f_1(n)$ indicates the network may loose some flat maxims, when the a parameter is to small. Contrary to that we have to know the increasing of the a will cause the grow of the sensitivity of the network to the noise.

3. Neural network extended to the two dimensional case

By simple extending the network from the Fig.2 to the two dimensional case we get architecture presented on the Fig.6. It consists of two subsystems working on two perpendicular directions X and Y respectively. The extended form of the weight distribution we may rewrite in the form (for the first and the second layer respectively in the X direction):

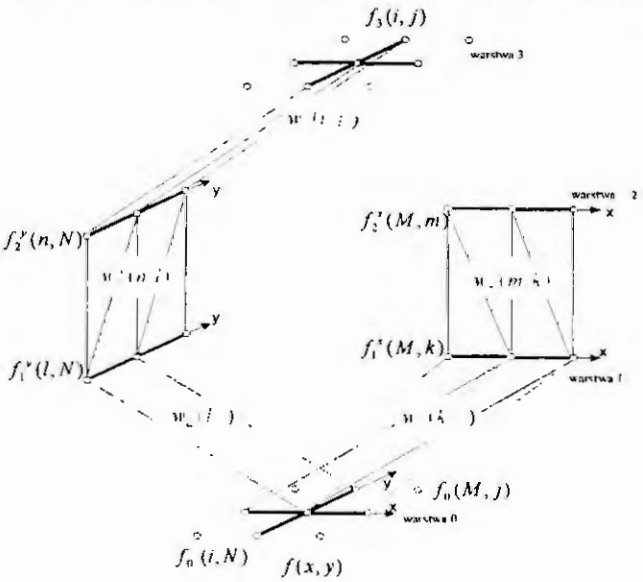


Fig.6 The network for image analysing.

$$w_a^x(k,j) = \begin{cases} a(-1)^{(j-k)}\binom{1}{k-j+1} & for \quad k,j \in [0, M_x - 1] \\ 0 & for \quad k,j \notin [0, M_x - 1] \end{cases}$$

and:

$$w_b^y(n,l) = \begin{cases} b(-1)^{(l-n)}\binom{1}{n-l+2} & for \quad l,n \in [0, M_y - 1] \\ 0 & for \quad l,n \notin [0, M_y - 1] \end{cases}$$

The additional third layer accumulate signals from the both directions so its weights might be written as:

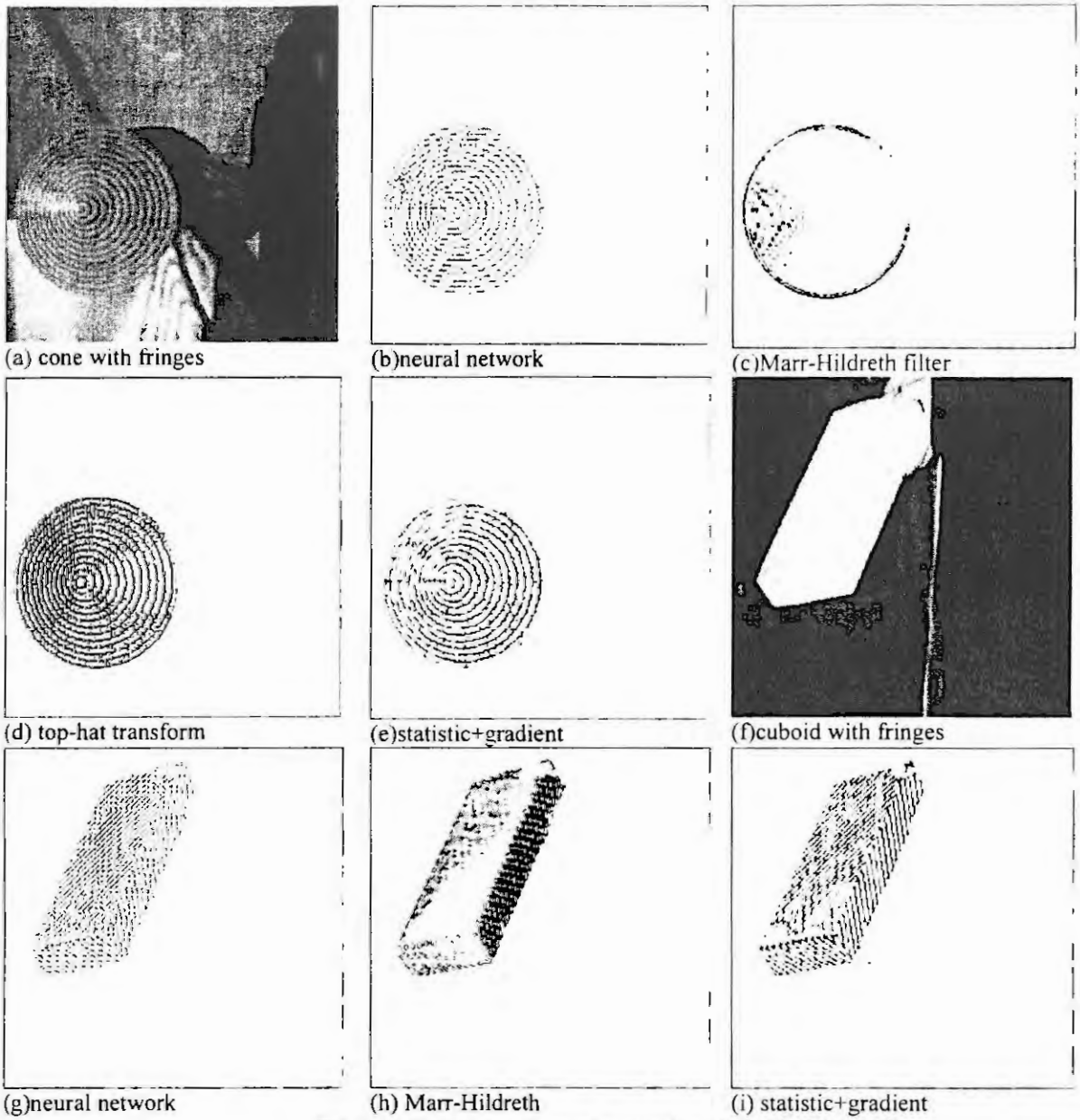
$$w_c(i,j') = c[[I][I]] = c\begin{bmatrix} 1 & 0 & 0 & \dots & 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1_{\{i,j=M_x-1\}} & 0 & 0 & 0 & \dots & 1_{\{i,j=2(M_x-1)\}} \end{bmatrix}$$

where c denotes positive weight coefficient of the output layer. The input signal to the neuron in this layer we may express as:

$$e_3(i,j) = \sum_{j'=0}^{2(M_x-1)} w_c(i,j') f_2(j',j) \tag{9}$$

4. Results of the experiments

Huge amount of experiments provided proof the prepared network perform this kind of interferometry images good. We made comprehensive comparison to the classical method of convolution and skeletoning which seems to be another positive test of this net. Several examples of the input images and obtained results have been shown below.



(a) cone with fringes (b)neural network (c)Marr-Hildreth filter

(d) top-hat transform (e)statistic+gradient (f)cuboid with fringes

(g)neural network (h) Marr-Hildreth (i) statistic+gradient

Fig. 7. Performance of neural network and classical methods

The quality of the obtained ridges map we may evaluate by watching on their wide. The wider line and more interruption the smaller location quality is.

References:

[Edw91[1]] T.Edwords, "Discrete Wavelet Transforms: Theory and Implementation", Standford University, September 1991, http:\\Intenet:www.sinh.stanford.edu\ wavelab.html

[Vida96[2]] B.Vidakovic, P.Mueller, "Wavelets for kids", Duke University, http:\\Intenet:www.sinh.stanford.edu\ wavelab.html, 1996.

[Stan96[3]] „Wavelab toolbox for use with Matlab at Stanford University", http:\\www.wavelab.sinh.stanford.edu, page supervised by Tim Edwards, Jonathan Buckheit and David Donoho

[Hsi97[4]] J.W.Hsiech, H.M.Liao, K.Fan, M.Ko, Y.Hung, „Image Registration Using a New Edge-Based Approach", Computer Vision and Image Understanding, vol.67, No.2, 1997r

[Hea98[5]] M.heath, S.Sarkar, T.Sanocki, K.Bowyer, "Comparison of Edge Detectors- Methodology and Initial Study", Computer Vision and Image Understanding, Vol.69, No.1 January 1998.

[Dud96[6]] E. Dudek-Dyduch, Z.Gomółka, R.Pękala "Adjusting parameters of two dimensional neural network", Proc. Of the Second Conference of Polish Neural Society, Szczyrk 1996.

[Gom99[7]] Z.Gomółka, „Neural Networks dedicated to edges detection" , 3rd International Modelling School Crimea'99, September 1999, Ukraine.

CLINICAL TRIAL SIMULATION AND DESIGN WITH BINARY OUTCOME DATA: THE NARATRIPTAN CASE STUDY

I. Nestorov[1], S. Duffull[2], L. Aarons[2], E. Fuseau[3], P. Coates[3]

[1] Centre for Applied Pharmacokinetic Research, School of Pharmacy, Manchester University, Oxford Road, Manchester M13 9PL, United Kingdom

[2] School of Pharmacy, Manchester University, Oxford Road, Manchester M13 9PL, United Kingdom

[3] Clinical Pharmacology, Glaxo Welcome Research and Development Ltd., Greenford, Middlesex UB6 0HE, United Kingdom

Abstract. In order to assess the benefits and potential of clinical trial simulation and design, a retrospective study with an anti-migraine drug naratriptan was carried out. The aim of the study was to simulate and design a phase II oral dose ranging clinical trial, which has the highest probability of showing the dose – response relationship for the drug. Using two alternative models – a PD model and a PK-PD model – extensive Monte-Carlo simulations were carried out to determine the influence of the controllable clinical trial variables (sample size, sampling and administration schedules) and the uncontrollable clinical trial variables (modelling assumptions, parameter and structure uncertainties, etc.) on the power of the trial. The results clearly demonstrate the potential of the clinical trial simulation for the quantitative assessment of the influence of the respective factors. A D-optimal design with respect to the doses to be administered and the effect sampling times leads to the selection of a more rational trial design, compared to the one that was actually carried out, needing less resources.

Introduction

There is recent and growing interest in using computer simulations of clinical trials to improve clinical trial design [1, 2]. In order to assess the benefits and potential of clinical trial simulation and design, a retrospective study with an anti-migraine drug – naratriptan – was carried out. The aim of the study was to simulate and design a phase II oral dose ranging clinical trial, which has the highest probability of showing the dose – response relationship for the drug. Pharmacokinetic (PK) and pharmacodynamic (PD) data from preceding clinical trials were available for analysis. During the trial simulations and design we were blinded, i.e. unaware of the outcomes of the actual trial that was carried out. This data was made available later for validation purposes.

Naratriptan data models

The phase I data include plasma concentration profiles after intravenous, subcutaneous and oral routes of administration from 26 healthy male volunteers. The Phase IIa data are after placebo or subcutaneous administration to about 400 patients (mostly female), including 33 patients on active treatment with both plasma concentration profiles and headache score data.

The limitations imposed on the clinical trial design were:

1. Headache score is measured on a categorical scale with 4 categories: 3 - severe, 2 - moderate, 1 - mild, 0 - none.
2. Headache relief is defined by a change from categories 3 or 2 into categories 1 or 0, i.e. it can be modelled as a binary variable.
3. Headache relief is to be assessed at 2 hours and at other relevant times.
4. The trial should be parallel with a maximum 7 arms and a maximum total of 1500 subjects.

As the naratriptan effect is a binary variable, it is most conveniently modelled by a logistic regression model, which generates failures and successes (0's and 1's) with a probability *Pr* as shown at Fig. 1. Based on the population analysis of the data from Phase I and Phase IIa, two models of "the biology" of the systems were developed:

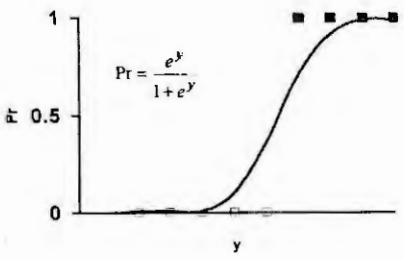


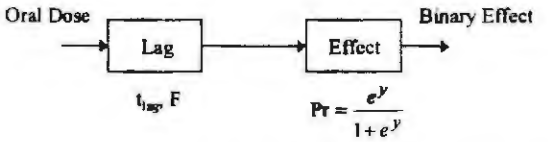
Fig. 1. Logistic regression model used to describe the Headache relief from naratriptan.

PD model: (Table 1, Fig. 2) with parameters, identified from the Phase IIa data alone; the PK data were ignored.

PK-PD model: (Table 2, Fig. 3) with parameters estimated from Phase I and Phase IIa data. The predicted dose - effect curves for naratriptan using the PD and the PK-PD models are shown in Figures 4 and 5 respectively.

Table 1. Parameters of the PD model.

Parameter	Value	SD
tlag [h]	0.9	n.a.
F [%]	60	30
β0	-1.264	0.630
β1	0.543	0.270
β2	0.042	0.021
β3	0.164	0.082
β4	0.4	n.a.

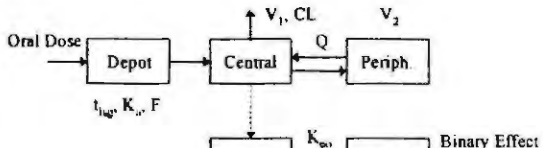


$$y = \theta + \beta_1 \times \text{time} + \beta_2 \times \text{dose} + \beta_3 \times \text{dose} \times \text{time} + \beta_4 \times \text{type}$$

Fig. 2. PD model for naratriptan.

Table 2. Parameters of the PK-PD model.

Param.	Value	SD	Param.	Value	SD
tlag [h]	0.35	0.35	Keo [1/h]	0.845	n.a.
F [%]	48	n.a.	β0	-0.11	0.06
Ka [1/h]	0.85	0.28	β1	0.482	n.a.
CL [L/h]	17.9	7.57	β2	0.082	n.a.
V1 [L]	23.00	20.1	β3	0.083	n.a.
V2 [L]	98.4	47.0	β4	0.714	n.a.
Q [L/h]	101.4	36.8			



$$y = \theta + \beta_1 \times \text{time} + \beta_2 \times C_e + \beta_3 \times C_e \times \text{time} + \beta_4 \times \text{type}$$

Fig. 3. PK-PD model for naratriptan.

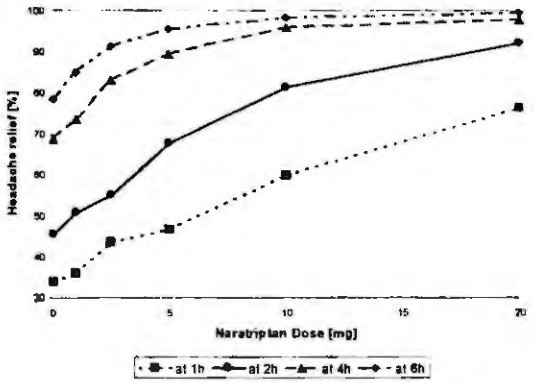


Fig. 4. Dose-effect profile of the PD model.

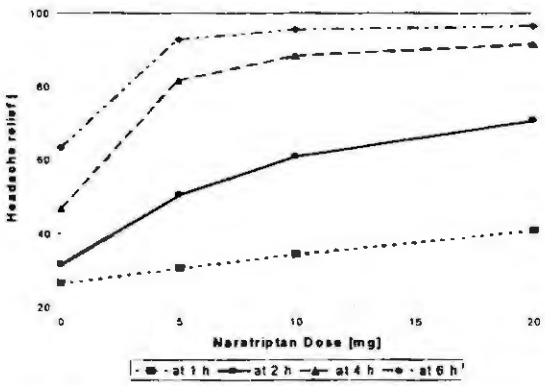


Fig. 5. Dose-effect profile of the PK-PD model.

Clinical trial simulation for naratriptan

As one of the primary aims of the study was to design a phase II oral dose ranging trial, which has the highest probability of showing the dose - response relationship for naratriptan, one should concentrate on the power of the trial designed, i.e. the ability of the trial to detect statistically and clinically distinct points of the dose-effect curve. Both the PD and the PK-PD model were used in extensive Monte-Carlo simulations, testing the effect of

the sample size per arm, the number of subjects in the placebo arm, the response sampling times and some of the modelling assumptions (bioavailability variance, lagtime and Keo variance), on the power of the trial to detect a statistically significant difference in the outcome. In each of the above cases, 200 trials were simulated, each trial had 4 doses (arms) - placebo, 5 mg, 10 mg and 20 mg naratriptan.
A comparison of the power of the trial with different sample sizes (50, 100, 150 and 200 subjects per dose) at 2 h showed that, even with the PK-PD model, which seems to be more conservative, a sample size of 200 subjects per arm (dose) would guarantee a power, greater than 80%. An interesting observation is that a much smaller sample size for the placebo arm may be needed. Even only 25 subjects in the placebo arm (and 125 in each of the other three) ensures more than 90% power of the trial.
The simulation of the influence of the Headache relief sampling time on the power of the trial shows that (with the 2 h time point required by the regulator) the power of the trial would be maximised if 4 and/or 6 h are included in the sampling schedule too.
It is well known that migraine is usually accompanied by delayed gastric emptying, which would increase the PK lag-time of the system. Yet, the data available did not provide information about the magnitude of the PK lag, as it was derived from the data for an oral formulation, administered to healthy subjects only. The available Phase I PK data were for an oral solution, which lead to a clear underestimation of the true lag time to be expected with a tablet. Therefore, the influence of the lag time on the power of the trial had to be explored by simulation. The results with lagtimes of 0.5 h and 1 h show that it is realistic to expect that for most doses the proposed trial would perform well, with a power above 80%, if the PK lagtime of naratriptan is in this range.
The experimental data also show evidence of a significant variability in the bioavailability of naratriptan, the likely CV being in the region 40 - 60%. Yet, the population model used for the PK-PD model was unable to estimate this variability, possibly assigning it to the absorption constant *Ka* (Table 2). The same refers to the link model rate constant *Keo*, which represents the PD delay in the system. In order to study the influence of the model parameter uncertainties on the power of the trial, a clinical trial with different CV's (0, 30% and 60%) for the parameters of interest was simulated. The results show that the influence of both parameters is similar, but even in the worst case of a CV above 30%, the power of the clinical trial seems to be sufficient.
The sample size can be calculated alternatively using the output of the models, without doing any simulations. For this purpose, a number of statistical tables are available in the literature [3], using different approximations of the binomial distribution. The tables give a conservative estimate of the sample size needed, compared to the simulations results, due to the normal approximation of the binomial distribution used.

D-optimisation of the trial design for naratriptan

The list of doses and the effect sampling times can also be determined solely on the basis of the models developed, without any clinical trial simulation, using Design of Experiments (DOE) techniques. The dose ranging study can be redefined in terms of a dose - effect model parameter estimation study and a D-optimisation experimental design procedure can be applied, aiming at the minimisation of the uncertainty in the dose - effect model parameters estimates to be derived. D-optimal designs maximise the Fischer Information Matrix and by that minimise the confidence regions of the parameter estimates [4]. If the type of the migraine in both the PD (Fig. 2) and PK-PD (Fig. 3) models is ignored, then the response model has 4 parameters to be estimated (θ, β_1, β_2, and β_3) and two covariates - time and dose (for the PD model) or plasma concentration (PK-PD model). As the number of design points of the optimal design is equal to the number of parameters estimated, the optimal design in our case includes two doses and two effect sampling times. The additional constraints to the sets of possible doses and effect sampling times are: (*i*) they are positive; (*ii*) a placebo dose is required; and (*iii*) a 2h effect sampling time is required by the regulator.
Taking into account the above considerations, we extended the DOE theory to our case of binary response data with two covariates and an interaction term in the logistic function. Subsequently, a D-optimisation procedure, programmed in MATLAB was implemented to determine the optimal design points. The Fischer information matrix surface, which was computed with two fixed points of the design - a placebo dose and an effect sampling point at 2 h showed that a maximum is attained at the early sampling times (close to 0 h) and at a second dose of around 10 mg. For the sake of the balance between logistics and optimality, we selected as a suboptimal design the one with a second dose of 10 mg and a second sampling time of 1 h.
To compare quantitatively the alternative trial designs, which are more logistically realistic, their relative efficiencies, with respect to the suboptimal design, were calculated from the respective Fischer information matrices [4], and are shown in Table 3. In terms of D-optimality, the proposed designs have close to or even better efficiencies than the design which was applied and published [5]. At the same time, the proposed designs include only half the number of the doses. The latter shows the need for a careful balance between the design optimality and the logistics in order to achieve the best trial results.

Table 3. Relative efficiency of several possible clinical trial designs for naratriptan.

Doses [mg]	Effect Sampling Times [h]	Replicates	Number of points	Determinant of the Fischer Matrix	Relative efficiency [%]	Note
0, 10	**1, 2**	**4**	**16**	**637**	**100**	Suboptimal design
0, 2.5, 10, 20	**2, 4**	**2**	**16**	**33.3**	**47.8**	Proposed design 1
0, 1, 2.5, 10	**2, 4**	**2**	**16**	**75.0**	**58.6**	Proposed design 2
0, 1, 2.5, 5	**2, 4**	**2**	**16**	**115**	**65.2**	Proposed design 3
0, 0.1, 0.25, 1, 2.5, 5, 7.5, 10	**2, 4**	**1**	**16**	**107**	**63.9**	Applied design [5]

Clinical trial design for naratriptan

Based on the above simulations and theoretical considerations, a clinical trial with four or five arms with doses: placebo, 2.5 mg and/or 5 mg, 10 mg, and 20 mg was recommended. The simulations show that the minimum effective dose is approximately 1 mg, the latter dose may be included in the design, if necessary (the largest dose can be removed in this case). The simulations show that the maximum no-effect dose is in the range 0.5 - 0.75 mg. At least 150-200 subjects per arm are necessary if the required power of the trial is 80%. The Headache relief should be monitored at 2 h, 4h and/or 6 h.

Conclusions

The analysis of the results from the naratriptan clinical trial simulation and design shows that the following conclusions can be made:

1. Clinical trial simulation is a useful tool for quantitative assessment of the influence of the controllable clinical trial variables (sample size, sampling and administration schedules) on the power of the trial.
2. Clinical trial simulation is ***the only*** tool for quantitative assessment of the influence of the uncontrollable clinical trial variables (modelling assumptions, parameter and structure uncertainties, etc.) on the power of the trial.
3. Clinical trial simulation, combined with design of experiment techniques (e.g. D-optimal design), is a powerful instrument for rational clinical trial design.
4. Clinical trial simulation and design as a process is not very demanding in terms of time, software and computational power, but requires a significant knowledge and skill base.
5. Clinical trial simulation and design poses interesting fundamental research issues e.g. in the DOE, mixed effect modelling and sensitivity analysis areas.

The overall balance of the resources for and benefits from clinical trial simulation and design show that it has a significant and so far not fully utilised potential in the drug development process.

References:

1. Peck, C.C., and Desjardins R.E., Simulation of clinical trials: encouragement and cautions, Appl.Clin.Trials 5(1996): 30-32.
2. Hale, M., Gillespie, W.R., Gupta, S.K., Tuk, B., and Holford, N.H., Clinical trial simulation: streamlining your drug development process, Appl.Clin.Trials 5(1996): 35-40.
3. Lemeshow, St., Hosmer Jr., D.W., Klar, J., and Lwanga, St. K., Adequacy of sample size in health studies. WHO, New York, 1990.
4. E. Walter, E., and Pronzato, L., Qualitative and quantitative experiment design for phenomenological models - a survey. Automatica, 26(1990): 195-213.
5. Fuseau, E., Kempsford, R., Winter, P., Asgharnejad, M., Sambol, N., and Liu, C.Y., The integration of the population approach into drug development: A case study, Naratriptan. In: The population approach: measuring and managing variability in response, concentration and dose. (Eds: L. Aarons et al..) COST B1 Medicine. European Commission, Brussels, 1997, 203-214.

COMBINATION OF MODELING IN FREQUENCY AND TIME DOMAIN IN SURROGATE ENDPOINT EVALUATIONS

L. Dedík[1] and M. Durišová[2]
[1]Faculty of Mechanical Engineering, Slovak University of Technology
Námestie slobody 17, SK-812 31 Bratislava, dedik@kam1.stuba.sk
[2]Institute of Experimental Pharmacology, Slovak Academy of Sciences
Dúbravska cesta 9, SK-842 16 Bratislava, exfamadu@savba.sk

Abstract. A procedure based on a combination of system modeling in the frequency and time domain was introduced in our study: Durišová, M., *et al.*, Bull. Math. Biol., 57 (1995), 787-808. This procedure was designed to build structured models of linear dynamic systems, decomposable to submodels of individual subsystems. In the present study, the procedure is utilized in building models of systems describing dynamics of migration of lymphoid cells between blood, lymphatic system, and non-lymphoid tissue. The model selected as optimal to describe this process in Merino ewes under physiological conditions consisted from two submodels of first-order linear dynamic subsystems connected in serial and a submodel containing time delays in several parallel branches. This model allowed to identify and quantify several fractions in the total population of lymphocytes, obeying different migration dynamics. The modeling procedure utilized in the present study and the model selected could be used in investigations of the physiology needed to develop and evaluate lymphoid cell biomarkers and surrogate endpoints and to support their biological and clinical plausibility.

Introduction

Biomarkers are characteristics indicating biological or disease processes, while surrogate endpoints are biomarkers intended to substitute for clinical endpoints. At present, there is an increasing interest in the development of biomarkers and surrogate endpoints, with the goal to more efficiently evaluate new therapies in clinical research [1]. In this endeavor, it is a common way to utilize so-called pharmacokinetic/pharmacodynamic models (PK/PD), to look for linkages among disease progression, drug exposure, changes in biomarkers, clinical outcome, *etc.* A PK/PD model can be created *e.g.* by simultaneous fitting *a priori* known functions such as polyexponential and sigmoid functions to a drug concentration-time profile in blood of a subject and to a biomarker-drug concentration profile in this subject, respectively. This method can yield good fits to both profiles mentioned. In general, however, it cannot yield the mathematical model of the linkage between the drug concentration-time profile in blood and the biomarker-time profile of the subject allowing successful prediction of time profiles this biomarker from other time profiles of the concentration of the drug in blood in the same subject.

In contrast to this, considering from the system approach point of view, the linkage between the two time profiles can be represented by a dynamic system, defined in such a way that one of these profiles is considered the input and the other the output of this system. If this system is time invariant and can be sufficiently approximated by a linear model, modeling methods based on the system transfer function can be employed to identify a model of this system [2].

The migration of lymphocytes between blood, lymphatic system, and non-lymphoid tissue is crucial both to the surveillance of foreign antigen and to providing an effective local immune response to infection. The importance of the variability in the size of the blood population of the specific lymphoid cells (T cells) to migration factors is of increasing concern in the study of HIV infection [7]. The goal of our study is to build a structured model describing the linkage between the lymphocyte concentration-time profiles in the venous blood and in the prescapular lymph in Merino ewes under physiological conditions, *i.e.* a model describing the dynamics of the lymphocyte migration between blood, lymphatic system, and non-lymphoid tissue in these animals. Such a model could be used in investigations of the physiology needed to develop and evaluate lymphoid cell biomarkers and surrogate endpoints and to support their biological and clinical plausibility.

Material and Methods

In a biological experiment [6], six randomly selected healthy Merino ewes, 3-5 years old, of 25-35 kg weight, were used. Lymph was collected from the cannulated efferent lymphatic vessel of the prescapular lymph node of the animals. Lymphocytes were isolated and labeled with 3-5 μmol 5-(and 6)-carboxyfluorescein diacetate succinimidyl ester. The viability of the lymphocytes was tested by propidium iodide staining and analyzed by the florescence activated cell analyzer (FACScan), Becton Dickinson, CA. The labeled lymphocytes were injected back into the animals via an indwelling venous cannula. The concentration-time profiles of the labeled lymphocytes in the venous blood and prescapular lymph were determined using the FACScan.

In our study, the dynamic system representing the lymphocyte migration between the venous blood and prescapular lymph was defined by the transfer function in the Laplace domain. The concentration-time profile of the labeled lymphocytes in the venous blood was considered the input of this system, while the concentration-time profile of the labeled lymphocytes in the prescapular lymph was considered the output of this systems. The software package CXT-MAIN (Fig. 1) was used to model this system in the frequency domain [2-4].

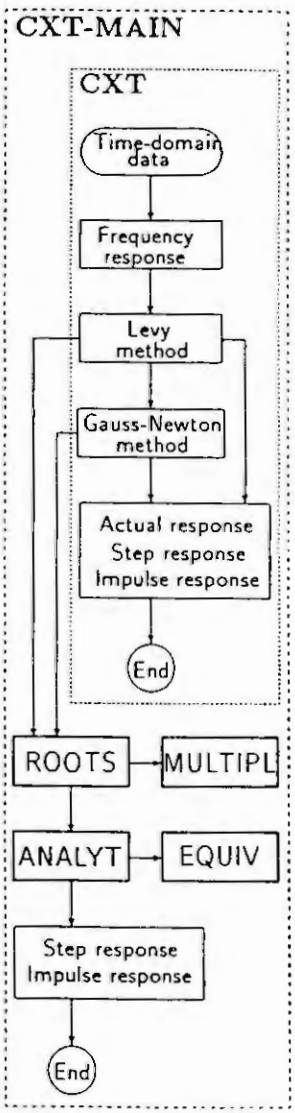


Fig. 1. CXT-MAIN software package. The package employs the model frequency response $F_M(\omega)$ in the form of Eq. 1,

$$F_M(\omega) = G_M \frac{a_0 + a_1 i\omega - a_2\omega^2 + \ldots + a_n(i\omega)^n}{1 + b_1 i\omega - b_2\omega^2 + \ldots + b_m(i\omega)^m}, \qquad (1)$$

where G_M is the gain, $a_0, a_1, a_2, \ldots, a_n$, $b_1, b_2, \ldots, b_m$ are the parameters, ω is the radial frequency, and i is the imaginary unit [2-4]. The Levy method [5] and the Gauss-Newton method are respectively used to obtain point estimates of the model parameters in the frequency domain and interval estimates of the model parameters (expressed by uncertainties of type A (ISO TN 1297)) in the time domain. The Euler method is employed to determine the system response to an actual input, and the system step and impulse responses. The CXT-MAIN has four complementary programs: *1)* ROOTS for the determination of real and imaginary (non-multiple and multiple) roots; *2)* ANALYT for the determination of the analytical form of the model weighting function; *3)* MULTIPL for the multiplication of two model transfer functions; *4)* EQUIV for testing dynamic equivalence of two dynamic models. The CXT-MAIN enables acquisition of accurate models of simple and/or complex linear dynamic systems, including systems with feedback, shunts, and time delays. The parameters of the non-structured model frequency response in the form of Eq. 1 generally do not permit direct physical interpretations, especially in the case of high-order model frequency responses. However, this model implicitly contains useful information about the system structure. Thus it can be considered an intermediate model between the system as represented by measurements and by its structured model. This can be very effectively employed in procedures for building structured models of the systems studied [4], consisting from the following main steps: *1)* Approximations of the system by auxiliary models in the time having different arrangements of models of the elementary systems [2], aimed at selection of model structures whose outputs and weighting functions are similar to the measured system output and the system weighting function determined by the model frequency response in the form of Eq. 1, respectively; *2)* Selection of the structured models possessing good statistical properties; *3)* Selection of the structured model yielding a minimum value of the Akaike information criterion.

The model frequency responses of the dynamic systems describing lymphocyte migration and the procedure published in our study [4] were used to identify structured models of these systems in the time domain, on taking into account *a priori* knowledge about the physiology of the lymphocyte migration process [6-7].

Results

The concentration-time profiles of the labeled lymphocytes in the venous blood and prescapular lymph, respectively, in the representative animal are shown in Figs. 2a and 2b. The general structure of the model,

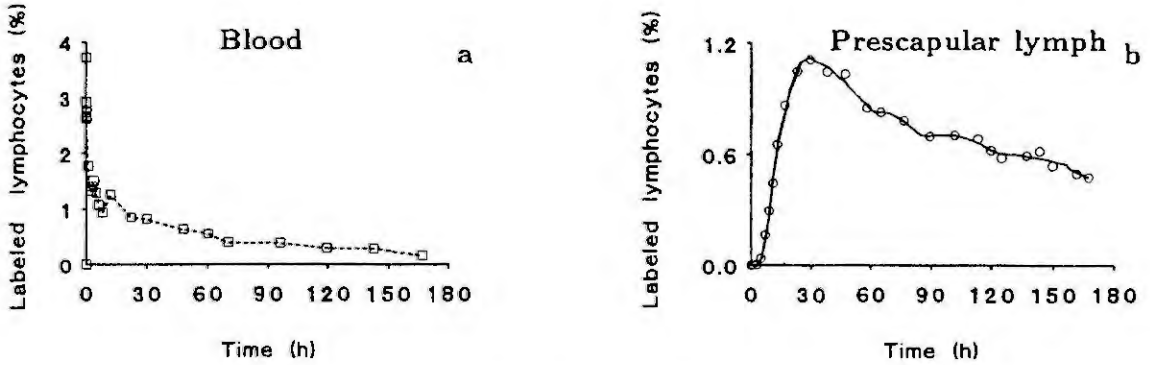


Fig. 2. a) Time profiles of the concentration of the labeled lymphocytes in blood (squares); b) Time profile of the concentration of the labeled lymphocytes in the prescapular lymph (circles) and the model (full line)

selected as optimal for the dynamic systems describing of the lymphocyte migration between the venous blood and the prescapular lymph in healthy Merino ewes under physiological conditions is shown in Fig. 3. It consists from the two submodels of first-order linear dynamic subsystems connected in serial having the time constants T_1 and T_2, and the submodel containing the gains G_j and the time delays τ_j, $j = 1,2,\ldots,k$ in k parallel branches. The transfer function of this model is given Eq. 2

$$H_M(s) = \frac{\sum_{j=1}^{k} G_j e^{-\tau_j s}}{(1+T_1 s)(1+T_2 s)}, \qquad (2)$$

where s is the Laplace variable. This model was determined using the definition of the dynamic system representing the lymphocyte migration between the venous blood and prescapular lymph by the transfer function given by Eq. 3

$$H(s) = \frac{C_L(s)}{C_B(s)}, \qquad (3)$$

where $C_L(s)$ and $C_B(s)$ are the Laplace transforms of the concentration-time profiles of the labeled lymphocytes in the prescapular lymph and in the venous blood, respectively. It follows then that this model can be used for predictions of concentration-time profiles of the labeled lymphocytes in the prescapular lymph from other concentration-time profiles of the labeled lymphocytes in the venous blood in the same animal, and *vice versa.*

The interval estimates of the parameters of this model for the representative animal were: $T_1 = 9.82 \pm 2.34h$, $T_2 = 4.88 \pm 0.09h$, $G_1 = 1.19 \pm 0.02$, $\tau_1 = 4.29 \pm 0.55h$, $G_2 = 0.15 \pm 0.05$, $\tau_2 = 60.71 \pm 3.97h$, $G_3 = 0.10 \pm 0.04$, $\tau_3 = 84.70 \pm 5.91h$, $G_4 = 0.08 \pm 0.02$, $\tau_4 = 122.90 \pm 7.73h$. The response of this model to the profile shown in Fig. 2a is illustrated in Fig. 2b. The model transfer function given by Eq. 2 allowed to derive formulas for the fractions $F_j = G_j / \sum_{l=1}^{k} G_l$ of the total lymphocyte population traveling through the branch j, for the mean transit times $MTT_j = T_1 + T_2 + \tau_j$ corresponding to the branch j, and for the MTT of the whole dynamic system

$$MTT = \frac{1}{\sum_{l=1}^{k} G_l} \sum_{j=1}^{k} (T_1 + T_2 + \tau_j) G_j . \qquad (4)$$

The estimation of the fractions F_i of the lymphocytes traveling through the individual branches yielded for the representative animal the values: $F_1 = 78.2\%$, $F_2 = 9.9\%$, $F_3 = 6.6\%$, and $F_4 = 5.3\%$ of the total population of the lymphocytes. The estimates of the mean transit times were: $MTT_1 = 18.9h$, $MTT_2 = 75.4h$, $MTT_3 = 99.4h$, and $MTT_4 = 137.6h$. The estimate of the mean transit time corresponding to the whole dynamic system describing the lymphocyte migration between the venous blood and prescapular lymph was $MTT = 36.1h$.

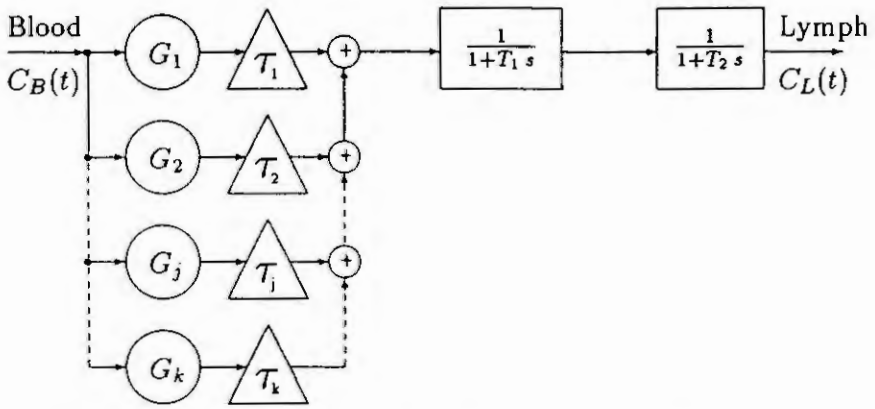


Fig. 3. Structured model describing dynamics of lymphocyte migration between blood and prescapular lymph

Conclusion

Models analogous to that presented above were identified for all the animals enrolled in study [6]. The greater was the time delay of a branch, the less was the lymphocyte fraction, while the reciprocal values of the lymphocyte fractions were linearly dependent on the time delays. The branches in the model shown in Fig. 3 correspond to the various lymphocyte inputs into the lymphatic/interstitial pool assumed in the abstract model presented in study [7]. However, in contrast to that model, the model in the form of Eq. 2 represents the real relationship between the lymphocyte concentration-time profiles in the venous blood and prescapular lymph, since it mathematically takes into account both these profiles. The procedure used, can be employed in building models of lymphocyte migration under physiological and/or pathological conditions.

Acknowledgments

This work was supported in part by Grant 1/4249/97 from the Slovak Grant Agency.

References

1. Book of Abstracts: Conference - Biomarkers and Surrogate Endpoints, Washington, April 15-17, 1999.
2. Dedík, L. and Durišová, M., System Approach in Technical, Environmental, and Bio-Medical Studies. Publishing House of the Slovak University of Technology, Bratislava, 1999.
3. Dedík, L. and Durišová, M., CXT-MAIN: a software package for determination of the analytical form of the pharmacokinetic system weighting function. Computer Methods and Programs in Biomedicine, 51 (1996), 183-192.
4. Durišová, M., Dedík, L., and Balan, M., Building a structured model of a complex pharmacokinetic system with time delays. Bulletin of Mathematical Biology, 57 (1995), 787-808.
5. Levy, E. C, Complex-curve fitting. IRE Transactions on Automatic Control, AC-4 (1959) 37-43.
6. Srikusalanukul, W., Modelling of Lymphocyte Migration in Peripheral Lymphoid Tissue, (Ph.D. Thesis). John Curtin School of Medical Research, Australian National University, Canberra, Australia, 1999.
7. Stekel, D. J., Parker, C. E., and Nowak, M. A., A model of lymphocyte recirculation. Immunology Today, 18 (1997) 216-221.

INTERSPECIES PHARMACOKINETIC MODEL VALIDATION & ALLOMETRY

J. F. Young[1] and R. H. Luecke[2]
[1]National Center for Toxicological Research
Jefferson, Arkansas 72079 USA
[2]University of Missouri-Columbia, Columbia, Missouri 65211 USA

Abstract. Not all data sets are adequate to validate even a simple pharmacokinetic model. Insufficient data might suggest a one compartment linear model when just a few more data points would require a two compartment model. Blood concentration data alone without knowledge of elimination characteristics handicap the model by making the dose and mass balance immaterial. Lack of data always makes "fitting" the data easier, but it does not always produce the "true" model and pharmacokinetic parameters. The environmental toxicant methyl mercury (MM) is slowly metabolized to inorganic mercury (IM) in all species. Most of the literature is based on radiolabeled ^{203}Hg whole body or whole blood assays. A simple two compartment model adequately fits almost all of this data, and a terminal elimination rate constant (β) can easily be determined. When extensive blood and tissue data over an extended time period are analyzed for both MM and IM, a larger and more complex physiologically-based pharmacokinetic (PBPK) model is required. However, such extensive data are not available for all species or even feasible from humans. Validating the MM/IM PBPK model across species using sometimes limited data sets and allometry will be discussed.

Introduction.

If the objective of an experiment is only to determine how fast a chemical is cleared from the blood of an animal, then blood concentration values alone are probably adequate. However, if you need to know where the chemical is going, then elimination data are needed. If you want to compare the fate of a chemical across species, then more complete models including elimination data are necessary. If incomplete recovery is obtained from the elimination data, then tissue data is necessary to complete the model of the fate of the chemical.

Assay characteristics are also important in model development. If the assay measures parent chemical only, model evaluation is simpler. If, as with the case of radiolabeled chemicals, the assay depicts parent chemical and metabolites as a single entity, the decay characteristics of a blood curve may be fit by a simple model which may not describe the true fate of the chemical. More complex models that describe both the parent chemical and metabolite are necessary which in turn require more detailed data from blood, tissues and elimination products.

Methyl mercury (MM) is an environmental toxicant which has long lasting effects on the brain, muscle, liver, and kidney [2]. Methyl mercury is distributed to all parts of the body and is slowly metabolized to inorganic mercury (IM) which in turn is distributed widely. Both entities are slowly eliminated from the body. Both entities are implicated in the toxicities.

An intricate multi-compartment physiologically-based pharmacokinetic (PBPK) model has been constructed [8] which simulates both the MM and IM simultaneously in all compartments of the body [9]. As the PBPK model was initially designed for use during pregnancy, mathematically defined growth functions are an integral part of the model [10,16] and allow the model to be applied to animals of various sizes.

Simulation of pharmacokinetic data from multiple species yields insight into the behavior of a xenobiotic that a single study or single species data can not offer. As more data sets are simulated, the model becomes more accurate and explicit. We have simulated MM and IM data sets from mouse, rat, guinea pig, cat, rabbit, monkey, sheep, pig, goat, cow, and human.

Methods.

The PBPK model used in these simulations has been previously described [8,9]. The four component model (i.e., four PBPK models in one; two chemicals in maternal and embryo/fetal systems) has been reduced to only utilize the two maternal parts of the model for MM and its metabolite, IM. The embryo/fetal portions of the pregnancy model are not utilized as all data simulated are for adult, non-pregnant animals. However, some growth functions were still necessary to fit some of the data.

The fit of each data set is based in principle on the rat data taken from Farris et al. [3] which does involve adult weight gain and has been previously reported [9]. Initially, only two parameters were adjusted in order to fit the various data sets; i.e., the metabolic conversion from MM to IM, and the overall tissue binding capacity.

Data was extracted from figures from the various primary literature articles by scanning (PaperPort for

WorkGroups, Version 2.0 for Windows, Visioneer, Palo Alto, CA) the published graphs and saving them as PCX files. These files were in turn imported into UN-SCAN-IT (Silk Scientific, Inc., Orem, Utah) where the data points were located, (x,y) coordinates were obtained, and concentration-time data was estimated. The concentration-time data was then used as input values for the PBPK program.

Results.

When human volunteers consumed fish containing four different natural levels of MM contamination 3-4 times per week for about three months, the decay curve obtained from the total mercury (Hg) analysis of the blood can be described with a single compartment model with an elimination halflife of 49.5 days (Figure 1) [14]. One could deduce from this data that mercury would be cleared from the body slowly, but with time would be reduced to a low and probably insignificant level.

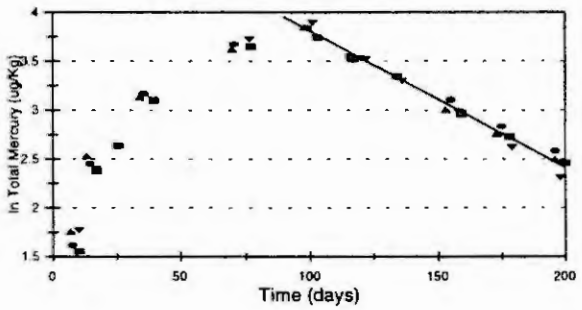


Figure 1: Total mercury blood levels from 20 human volunteers [14]. The different symbols represent different exposure levels of MM normalized to one dose level. The slope of the regression line for the elimination phase indicates an halflife of 49.5 days.

However, this is not the complete picture. Methyl mercury is metabolized to IM and both parent compound and metabolite are extensively and differentially distributed throughout the body. This is best illustrated with the rat data of Farris et al. [3] in Figure 2. In this study rats were serially sacrificed, and numerous tissues were analyzed for both MM and IM throughout an extended time period. The IM in the blood is about two orders of magnitude lower than the MM in blood; however, the IM in the kidney is an order of magnitude larger than the MM in blood and has a shallower terminal slope. This would indicate that IM will remain in the body long after blood Hg monitoring would indicate a clean system.

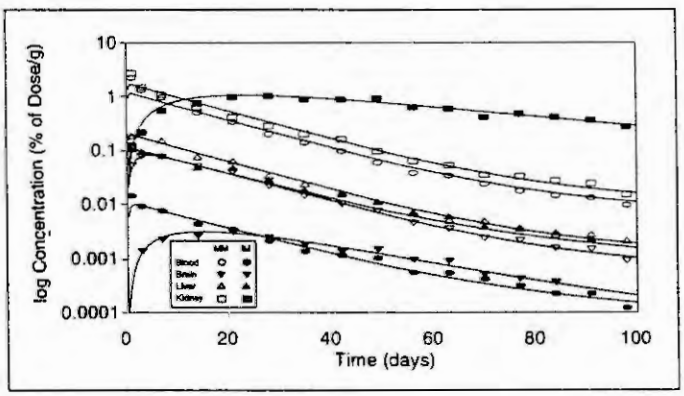


Figure 2: MM (open symbols) and IM (closed symbols) concentrations in blood, brain, liver, and kidney tissues of rats [3]. Simulated curves were generated from the PBPK model of Luecke et al. [9].

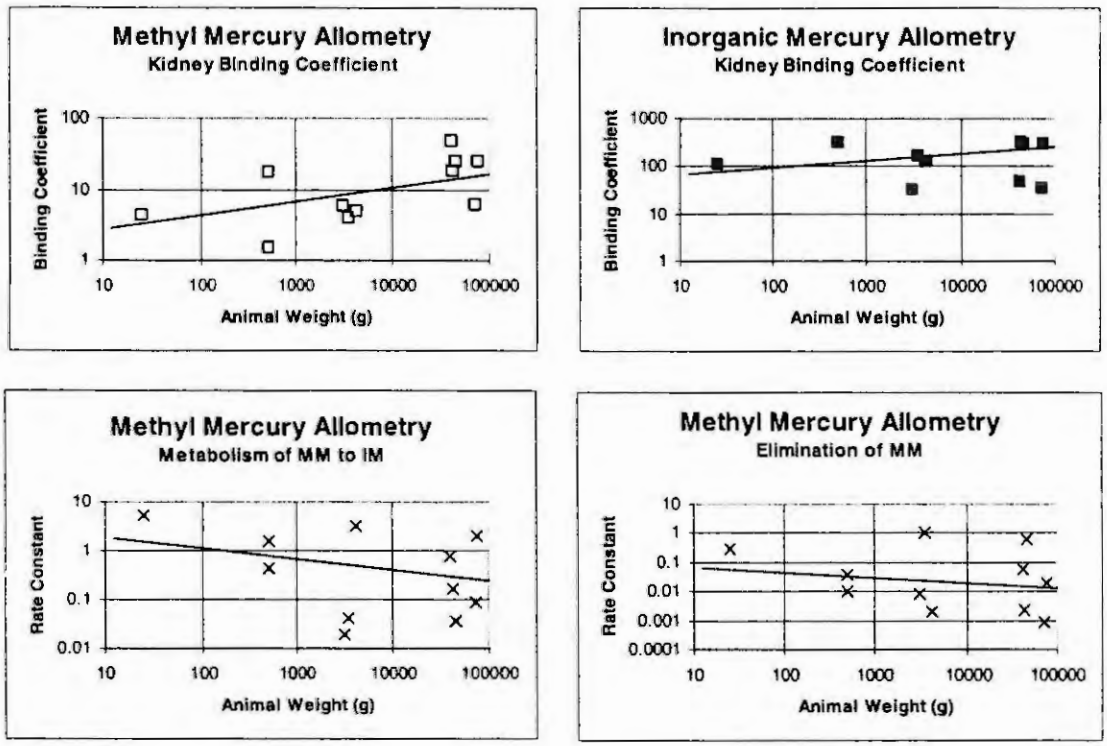


Figure 3: Allometric relationships for binding coefficients and rate constants.

Using this same PBPK model, data of varying complexity and differing times following MM administration were simulated from mouse [11], guinea pig [6], cat [5], rabbit [12], monkey [15], sheep [7], pig [4], goat [13], and cow [1,13]. Allometric relationships for the kidney binding coefficients (MM and IM) and metabolic (MM -> IM) and elimination (MM) rate constants are given in Figure 3. The IM binding coefficient for the kidney is about an order of magnitude larger than the MM kidney binding coefficient. This same magnitude and ratio holds for the liver. The brain binding coefficient is about an order of magnitude lower than the kidney and liver for MM, and about two orders of magnitude lower for IM. All of the binding coefficients have a positive allometric slope indicating that humans have a larger binding capacity than the smaller experimental animals. The metabolism of MM to IM is about 10 times faster than the elimination of MM. The negative slope indicates that humans convert MM to IM slower than the smaller laboratory animals and also eliminate MM slower. Together these relationships indicates that humans will retain mercury longer than smaller animals.

Utilizing all of this data and re-evaluating the data of Sherlock et al. [8], the fit of the PBPK model to the total

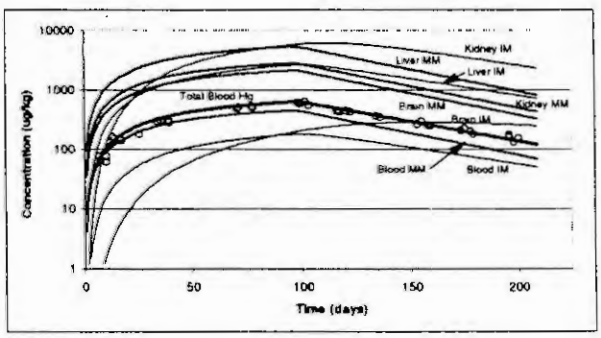


Figure 4: PBPK simulation of the total mercury human blood data of Sherlock et al. [8]. Also included are the MM and IM simulated curves for kidney, liver, brain, and blood.

mercury level in the blood indicates that all tissue levels are higher than the blood by the end of the experimental period (Figure 4), some by more than an order of magnitude.

Discussion.

Physiologically-based pharmacokinetic models can be utilized to simulate data from various species including man. However, choosing and validating the model requires sufficient data to assure the uniqueness of the model; i.e., a sparse data set can be fit with almost any model! Even the richness of the Farris et al. [7] data set has been fit with at least two models which are similar but not exactly the same [4,7].

The MM pharmacokinetic literature is fairly large with over 230 articles (13 species) in our personal data base. However, usable data that includes speciation, tissue concentrations, and data from extended sampling to define the decay characteristics are limited to only a few articles. Even the thoroughness of the Farris et al. [7] work would have benefitted from even longer time sampling in order to confirm the kidney and brain levels of inorganic mercury.

The discriminating features of this methyl mercury/inorganic mercury PBPK model are (1) the conversion of MM to IM, (2) the binding coefficients for both chemicals into kidney, liver, and brain tissues, and (3) the elimination of MM and IM. Of these the first seems to be the most critical and is the determinate step in the elimination of methyl mercury.

The lack of human data that has both organic and inorganic mercury concentrations over an extended time frame makes validation of the model problematic. However, when utilizing various human autopsy studies, IM pharmacokinetic studies, and in concert with MM pharmacokinetic studies and allometry, a viable model validation scheme can be presented. Utilizing all of the above data, the accumulation and decay data of Sherlock et al. [8] can be simulated for a validation of the PBPK model to humans. This simulation would suggest that at extended times when the whole blood Hg concentration approaches zero, there is still a substantial amount of the original dose of MM retained in the body in the liver, brain, and kidney as both organic and inorganic mercury.

References.

1. Ansari, M.S., Miller, W.J., Gentry, R.P., Neathery, M.W. and Stake, P.E., Tissue ^{203}Hg distribution in young Holstein calves after single tracer oral doses in organic and inorganic forms. J. Anim. Sci., 36 (1973), 415-419.
2. Clarkson, T.W., The toxicology of mercury. Crit. Rev. Clin. Lab. Sci., 34 (1997), 369-403.
3. Farris, F.F., Dedrick, R.I., Allen, P.V. and Smith, J.C., Physiological model for the pharmacokinetics of methyl mercury in the growing rat. Toxicol. Appl. Pharmacol., 119 (1993), 74-90.
4. Gyrd-Hansen, N., Toxicokinetics of methyl mercury in pigs. Arch. Toxicol., 48 (1981), 173-181.
5. Hollins, J.G., Willes, R.F., Bryce, F.R., Charbonneau, S.M. and Munro, I.C., The whole body retention and tissue distribution of [^{203}Hg] methylmercury in adult cats. Toxicol. Appl. Pharmacol., 33 (1975), 438-449.
6. Iverson, F., Downie, R.H., Paul, C. and Trenholm, H.L. Methyl mercury: Acute toxicity, tissue distribution and decay profiles in the guinea pig. Toxicol. Appl. Pharmacol., 24 (1973), 545-554.
7. Kostyniak, P.J., Pharmacokinetics of methylmercury in sheep. J. Appl. Toxicol., 3 (1983), 35-38.
8. Luecke, R.H., Wosilait, W.D., Pearce, B.A. and Young, J.F., A physiologically based pharmacokinetic computer model for human pregnancy. Teratology, 49 (1994), 90-103.
9. Luecke, R.H., Wosilait, W.D., Pearce, B.A. and Young, J.F., A computer model and program for xenobiotic disposition during pregnancy. Comp. Meth. Prog. Biomed., 53 (1997), 201-224.
10. Luecke, R.H., Wosilait, W.D. and Young, J.F., Mathematical representation of organ growth in the human embryo/fetus. Int. J. Bio-Med. Comput., 39 (1995), 337-347.
11. Mehra, M. and Choi, B.H., Distribution and biotransformation of methyl mercuric chloride in different tissues of mice. Acta Pharmacol. Toxicol., 49 (1981), 28-37.
12. Petersson, K., Dock, L., Soderling, K. and Vahter, M., Distribution of mercury in rabbits subchronically exposed to low levels of radiolabeled methyl mercury. Pharmacol. Toxicol., 68 (1991), 464-468.
13. Sell, J.L. and Davison, K.L., Metabolism of mercury, administered as methylmercuric chloride or mercuric chloride, by lactating ruminants. J. Agric. Food Chem., 23 (1975), 803-808.
14. Sherlock, J., Hislop, J., Newton, D., Topping, G. and Whittle, K., Elevation of mercury in human blood from controlled chronic ingestion of methylmercury in fish. Human Toxicol., 3 (1984), 117-131.
15. Vahter, M., Mottet, N.K., Friberg, L., Lind, B., Shen, D.D. and Burbacher, T., Speciation of mercury in the primate blood and brain following long-term exposure to methyl mercury. Toxicol. Appl. Pharmacol., 124 (1994), 221-229.
16. Wosilait, W.D., Luecke, R.H. and Young, J.F., A mathematical analysis of human embryonic and fetal growth data. Growth Dev. Aging, 56 (1992), 249-257.

POPULATION PHARMACOKINETIC MODELS: PARAMETRIC AND NONPARAMETRIC APPROACHES

R Jelliffe[1], A Schumitzky[1], M Van Guilder[1], X Wang[1], and R. Leary[2]
[1]Laboratory of Applied Pharmacokinetics, USC School of Medicine, 2250 Alcazar St, Los Angeles, CA 90033, USA, (323)442-1300, jelliffe@hsc.usc.edu
[2]the San Diego Supercomputer Center, 9500 Gilman Drive, La Jolla CA 92903

ABSTRACT

Population modeling seeks to evaluate the contributions of interindividual and intraindividual variability, based on the raw subject data and the assay error, and to describe the findings in terms that are useful both for research and for optimal patient care. With parametric models, the probability distributions of each PK/PD parameter are described in terms of other parameters such as means and covariances which define the assumed shape of these distributions. Commonly used distributions are the normal or lognormal ones. The parameter values found are the single best estimates such as mean, median, or mode, which are felt to be the best estimators of the central tendency for each such distribution.

Nonparametric models have a different flavor. No such parametric assumptions are made about the assumed shape of a parameter distribution, nor is a single parameter value what is really sought. The approach proceeds rather from the point of view that the very best population model possible would be the correct structural model, plus the entire collection of each subject's exactly known parameter values, if it were somehow possible to know them. Nonparametric methods estimate essentially one set of parameter values for each subject, along with an estimated probability for each such set. The richness of the method is in the ability to obtain not simply a single estimate for the central tendency and one for the dispersion, but rather to estimate the entire population parameter joint density.

Optimal population modeling currently begins by determining the assay error pattern explicitly over its working range. One then uses a parametric population modeling approach to separate intra- from inter-individual variability. Having this information, one can then use a nonparametric approach to obtain the entire estimated population parameter joint density.

INTRODUCTION

Pharmacokinetic and dynamic population models provide the means to store past experience with the behavior of drugs, and to apply it to the care of future patients. They are used as the Bayesian prior to design the initial regimen for the next patient who appears to belong to the population in question.

As one makes a population pharmacokinetic model (see below) it is useful to search for relationships between the various parameters in the model and useful clinical covariates or descriptors, so that the model can then be reparameterized in terms of these descriptors, and dosage can be adjusted to this important and often changing clinical information. This provides the logical structure for precise dosage adjustment to body weight and renal function, for example, in order to achieve desired target serum or peripheral compartment concentrations. Using this approach, one then computes the regimen to achieve the desired target goals.

PARAMETRIC POPULATION MODELS

With parametric models, the various pharmacokinetic parameters themselves are described in terms of other single point parameter estimates such as measures of central tendency – means, medians, or modes, for example, and measures of dispersion - standard deviations or covariances. A traditional search has been for the best single point estimator of each parameter. Examples of such parametric population modeling approaches are the standard two-stage approach [1], the iterative two-stage Bayesian approach [2], the parametric EM method [3], nonlinear mixed effect modeling [4], and other variations on this approach [5,6]. Other population modeling approaches are the semi-nonparametric approach of Davidian and Gallant [7] and the hierarchical Bayesian approach of Wakefield and colleagues [8].

THE SEPARATION PRINCIPLE

The separation or heuristic certainty equivalence principle [9] describes control of a system when it is separated first, into obtaining single point parameter estimates for the model, and second, of using those single

point estimates to control the system. For most pharmacokinetic applications, controllers using this approach are suboptimal. This is a significant problem with current maximum aposteriori probability (MAP) Bayesian fitting and dosage design. The way around this problem is by incorporating improved approaches to dosage design which make use of the entire nonparametric population joint parameter density.

MORE ON PARAMETRIC POPULATION MODELS

In parametric population modeling, the probability distribution of the parameters is itself described by these other single-valued parameters such as means and covariances, for example. These other parameters impart an assumed shape to each pharmacokinetic parameter distribution, usually a Gaussian or lognormal distribution. In contrast, for nonparametric models, as we will see below, these parametric assumptions are relaxed. No assumptions at all are made about the shape of their probability distribution, except that it is the same for all subjects. This is what is meant by the terms parametric and nonparametric in this context.

An example of the parametric approach to population modeling is that of the iterative two-stage Bayesian (IT2B) method. One begins this approach by stating estimates of the initial means and standard deviations of the pharmacokinetic parameters. Based on these assumptions, the individual patient data are then examined, and each patient's MAP Bayesian posterior pharmacokinetic parameter values are determined as described above. This ends the first stage.

In the second stage, the population means and covariances are obtained from these individual subject parameter values [3]. These new population parameter means and standard deviations (SD's) are then used as the initial Bayesian priors, thus beginning a new iteration. Using these new priors, the MAP Bayesian posteriors are again found. Their population means and SD's are obtained. Again, these means and SD's are used as the Bayesian prior in still another iteration, and the MAP Bayesian posteriors are again found. This process continues iteratively until a convergence criterion is finally reached.

STRENGTHS OF PARAMETRIC MODELS

Parametric population modeling methods can separate variability between the various subjects from variability within the individual subjects. This is their main strength - the ability to separate inter- from intra-individual subject variability. If the inter-individual variability is large, there is significant diversity in the population with respect to the parameter values. If the intra-individual variability is large, it suggests that the patients were unstable or that significant noise and uncertainty existed in their therapeutic environment.

DETERMINE THE ASSAY ERROR EXPLICITLY

Frequently, assumptions are made about the shape of the assay error pattern, but relatively little attention is paid specifically to determining the magnitude of the true laboratory assay error pattern itself, over its entire working range. It is frequently assumed that the assay error is only a small part of all the other environmental errors and uncertainties. These other errors are due, for example, to errors in the amounts of the doses given, errors in recording when they were given, errors in recording when the serum samples were obtained, errors due to misspecification of the structural model, and errors due to changes in the pharmacokinetic parameter values during the study period.

However, it is not difficult at all to determine the assay error pattern specifically, over its entire working range, by measuring representative samples in replicate. For example, the blank sample always has a certain error. The assay may become somewhat more precise at low to middle concentrations, and then usually less so at higher concentrations. The form of the relationship between the measured concentration and the assay SD is nonlinear, and a polynomial has been a useful way to capture the nonlinarity in this relationship [10]. Use of such a polynomial to describe the assay SD permits each serum concentration data point be fitted by its Fisher information, the reciprocal of its variance.

START WITH A PARAMETRIC POPULATION MODEL

Once the assay error pattern and its polynomial has been carefully determined over the entire working range of the assay, a parametric population modeling method such as IT2B can be used. The remaining intra-individual variability can be described as a scaling factor with respect to the assay error. One can then determine what fraction of the total overall intra-individual variability is due to the assay error itself. This is very useful information. In the USC*PACK IT2B population modeling program, this factor of intra-individual variability is

called gamma. If gamma is 1.0, then there is no other source of error than the assay error itself. Usually gamma is larger, and ranges from about 2 to 4, showing that the assay error often ranges from about 1/2 to 1/4 of the overall intra-individual variability, a significant fraction.

A significant limitation is that most parametric population modeling methods in current use do not have the desirable property of statistical consistency. Consistency means that as the number of subjects becomes arbitrarily large, the population parameter estimates approach the true population parameter values [11]. Furthermore, parametric population models are often used only to provide a single value for each population parameter when developing dosage regimens for patients. Because of the separation principle, such regimens are usually suboptimal.

NONPARAMETRIC POPULATION MODELS

The nonparametric approach to population modeling was first introduced independently by Lindsay [12] and by Mallet [13]. They showed that the most likely parameter estimates are actually found to be in a discrete, not continuous, collection of sets of individual parameter values, each of which has a single value for each parameter, along with an estimate of the probability of that particular set of values. There is usually about one significant support point (set of parameter values) for each subject studied in the population. This approach makes no parametric assumptions (such as normality or unimodality) about the actual shape of the population parameter distribution. The shape of the distribution is determined solely by the population raw data itself. Means and covariances can easily be obtained. Nonparametric methods such as the nonparametric maximum likelihood (NPML) method of Mallet [13] and the nonparametric expectation-maximization (NPEM) method of Schumitzky [14,15] can discover, without additional help from covariates, unsuspected clusters or subpopulations such as fast and slow metabolizers of a drug. The nonparametric discrete collections of parameter values and their estimated probabilities reflects the fact that the ideal theoretical population model would simply be the entire collection of each subject's exactly known parameter values, if one could somehow know them.

In an examination of the capability of the nonparametric method, Figure 1 shows a carefully defined simulated population in which there was only one distribution for the volume of distribution (V) but two for the elimination rate constant (K), such as occurs for fast and slow metabolizers of a drug [15]. There was no correlation between the two parameters.

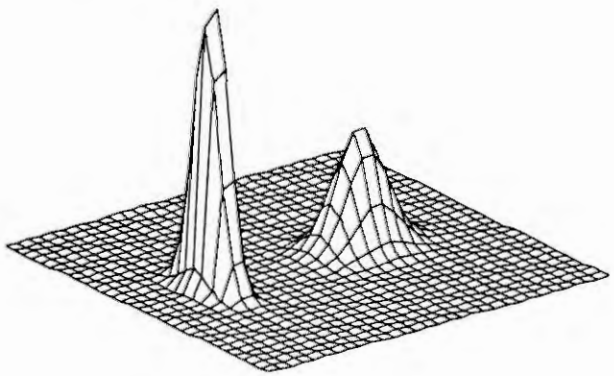


Figure 1. True population joint density of V and K. If the bottom corner is "home plate", then the axis toward third base is that of V, and the axis toward first base is that of K. Note that there are two subpopulations with respect to K, but only one for V. The vertical axis is the probability of a parameter pair v,k.

From this overall population, twenty subjects first were drawn at random. They constituted the sample population under study. Figure 2 shows the actual true distribution of these 20 subject parameter values. The task of any population analysis now is to discover this distribution optimally.

Figure 3 now shows the above true sample parameter distributions as estimated by the NPEM program. The NPML approach of Mallet is entirely similar. Two distinct and fairly tight groups of parameter values are seen. In contrast, Figure 4 shows these same true subject parameter distributions as estimated by a parametric population modeling method. A totally different, and erroneous, understanding of what is going on in the population is obtained by the parametric method. Where the mean value for the elimination rate constant actually exists, there are actually no subjects at all, as shown in Figures 1-3. The two groups of subjects were well

detected by the NPEM method, but were not detected at all by the parametric method. In addition, the nonparametric methods have the desirable property of mathematical consistency [11].

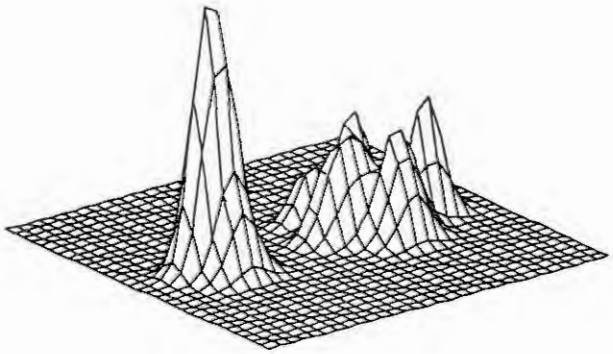


Figure 2. Graph, somewhat smoothed as in Figure 1, of the actual parameter values in the 20 sampled subjects. Axes as in Figure 1.

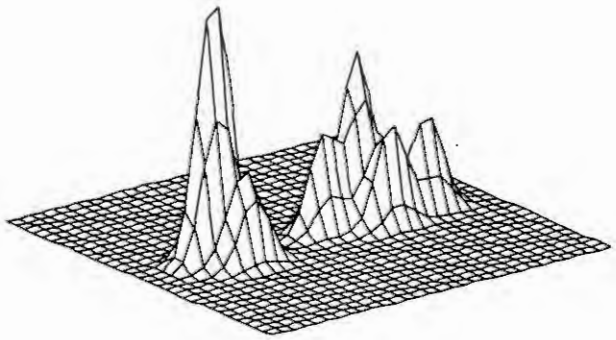


Figure 3. Estimated joint density obtained with NPEM. Axes as in Figure 1.

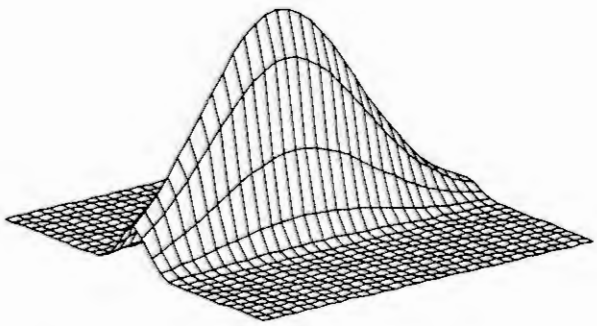


Figure 4. Estimated joint density obtained by an optimal parametric method. Axes as in Figure 1.

The nonparametric methods specifically obtain the single most likely distribution of parameter values for the population studied. This distribution is a multivalued distribution, and has a number of support points, or individual models, approximately equal to the number of subjects studied. The nonparametric approaches are superior to parametric methods in this respect, as they obtain the most likely representation of the entire population parameter distribution, while those based on parametric assumptions do not [13-15]. However, the nonparametric methods also have drawbacks. They cannot separate inter- from intra-individual sources of

variability. Further, they do not have confidence limits for the discrete distributions they obtain. Bootstrap methods are under consideration for developing such confidence limits.

CONCLUSION: USE BOTH METHODS SEQUENTIALLY

It currently seems that that the best approach to population modeling is to make use of both methods to take advantage of each of their individual strengths. First, it seems best to determine the specific assay error pattern and polynomial well, by itself, before beginning any analysis. If a multicenter study is being done, where different assays are being used in different settings, then each center should and can have its own assay error polynomial so that the subject data from each center can be analyzed correctly, each according to its own credibility. Next, gamma can be computed, using a parametric method such as the IT2B. In this way, one has obtained good knowledge of the intra-individual variability and of the assay error variability itself. Confidence limits on the parametric parameter distributions can also be obtained.

This information can then be used by the nonparametric approaches to obtain the entire, and most likely, discrete joint population parameter density. Software for IT2B and NP\EM population modeling approaches is available from our laboratory on PC's for 3 compartment linear models, and can also access the Cray T3E at the San Diego Supercomputer Center for larger and nonlinear models.

Acknowledgements: Supported by NIH grants LM 05401 and RR 11526.

References.

1. Rowland M, Sheiner L, and Steimer JL, eds. Variability in Drug Therapy: Description, Estimation, and Control. Raven Press, New York, 1985.
2. Sheiner L, Beal S, Rosenberg B et al.: Forecasting Individual Pharmacokinetics. Clin. Pharmacol. Therap. 26: 294-305, 1979.
3. Aarons L: The Estimation of Population Parameters using an EM Algorithm. Comput. Methods Programs Biomed. 41: 9-16, 1993.
4. Beal S, and Sheiner L: NONMEM User's Guide I: User's Basic Guide. Division of Clinical Pharmacology, University of California at San Francisco, 1979.
5. Lindstrom M and Bates D: Nonlinear Mixed-effects Models for Repeated Measures Data. Biometrics 46: 673-687, 1990.
6. Vonesh E and Carter R: Mixed Effets Nonlinear Regressions for Unbalanced Repeated Measures. Biometrics 48: 1-1, 1992.
7. Davidian M and Gallant A: The Nonlinear Mixed Effects Model with a Smooth Random Density. Biometrika 80: 475-488, 1993.
8. Wakefield J, Smith A, Racine-Poon A et al.: Bayesian Analysis of Linear and Nonlinear Population Models. Applied Stats. 43: 201-222, 1994.
9. Bertsekas D: Dynamic Programming: Deterministic and Stochastic Models. Prentice-Hall, Englewood NJ, 1987, pp. 144-146.
10. Jelliffe R: Explicit Determination of Laboratory Assay Error Patterns: a useful Aid in Therapeutic Drug Monitoring. No. DM 89-4 (DM56). Drug Monit. Toxicol. 10 (4): 1-6, 1989.
11. Spieler G and Schumitzky A: Asymptotic Properties of Extended Least Squares Estimates with Application to Population Pharmacokinetics. Proc. Am. Statistical Soc. Biopharmaceutical Section, San Francisco CA 177-182, 1993.
12. Lindsay B: The Geometry of Mixture Likelihoods: a General Theory. Ann. Statist. 11: 86-94, 1983.
13. Mallet A: A Maximum Likelihood Estimation Method for Random Coefficient Regression Models. Biometrika 73: 645-656, 1986.
14. Schumitzky A: Nonparametric EM Algorithms for Estimating Prior Distributions. App. Math. Comput. 45: 143-157, 1991.
15. Schumitzky A: The Nonparametric Maximum Likelihood Approach to Pharmacokinetic Population Analysis. Proceedings of the 1993 Western Simulation Multiconference: Simulation for Health Care. San Diego Society for Computer Simulation, pp. 95-100, 1993.
16. Van Guilder M, Leary R, Schumitzky A, Wang X, Vinks A, and Jelliffe R: Nonlinear Nonparametric Pharmacokinetic Modeling on a Supercomputer. Presented at the ACM/IEEE SC97 Conference, San Jose, CA, November 15-21, 1997.

MULTIPLE MODEL (MM) DOSAGE DESIGN: ACHIEVING TARGET GOALS WITH MAXIMUM PRECISION.

R Jelliffe, D Bayard, A Schumitzky, M Milman, F Jiang, S Leonov, and V Gandhi.
Laboratory of Applied Pharmacokinetics, USC School of Medicine, 2250 Alcazar St, Los Angeles CA 90033, USA. (323)442-1300, jelliffe@hsc.usc.edu

ABSTRACT

Most dosage regimens based on parametric population models as the Bayesian prior, including most Bayesian approaches of adaptive feedback control, use a single parameter value to describe the central tendency of each parameter distribution. Because of this, when a target goal is selected, the regimen to achieve it assumes that it does so exactly. However, the separation or heuristic certainty equivalence principle states that whenever a system is controlled, first, by obtaining single point parameter values, and then by using those values to control the system, the control achieved is usually suboptimal. In contrast, Multiple Model dosage design is based on nonparametric population models which have essentially one set of parameter values for each subject in the population. With this more likely Bayesian prior, multiple predictions are possible. Using these nonparametric models, one can compute the dosage regimen which specifically minimizes the predicted weighted squared error with which a desired target goal can be achieved. Other cost functions can also be employed.

As Bayesian feedback from serum concentrations is obtained, each set of parameter values in the nonparametric prior has its probability recomputed. Using this individualized nonparametric Bayesian posterior joint density, the new regimen to achieve the target with maximum precision is computed. In addition, a new Interacting Multiple Model (IMM) sequential Bayesian method has been developed to estimate such posterior densities when parameter values have been changing, as in unstable patients, during the time of analysis. A clinical software package implementing these approaches is in development.

INTRODUCTION: SET INDIVIDUALIZED TARGET GOALS FOR EACH PATIENT

The concept of a therapeutic range of serum drug concentrations is a generalization. It is an overall range in which most patients, but certainly not all, do well. One must always check each individual patient to see if he or she is doing not only well, but optimally, whatever the serum concentration is found to be. Figure 1 shows the usual means by which such therapeutic ranges are obtained. First, there is an increased incidence of therapeutic effects with increasing serum drug concentrations. Later on the incidence of toxic effects becomes significant. The eye is drawn to the upward bends in each line, and the classification of the therapeutic range is developed. However, this procedure does not deal with the need to develop a gentle dosage regimen for a patient who needs only a gentle touch, and a more aggressive one for a patient who really needs his dosage "pushed".

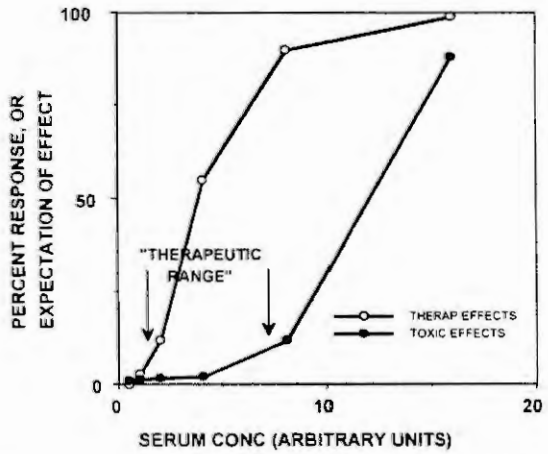


Figure 1. General relationships usually found between serum drug concentrations and the incidence of therapeutic and toxic effects. The eye is drawn to the bends in the curves, and the therapeutic range is classified in relation to these bends. This qualitative procedure of classification discards the important quantitative relationship of the incidence of toxic effects versus serum concentration.

Another approach is one in which the clinician evaluates each patient's individual clinical need for the drug in question, and selects an estimated risk of toxicity which is felt on clinical grounds to be justified by the patient's need. One then selects a target serum concentration goal to be achieved. One does not want the patient to run any greater risk of toxicity than is justified by the patient's clinical need. Within that constraint, however, one wants to give the patient as much drug as possible, to get the maximum benefit. This approach provides the rationale for selecting a specific target serum concentration goal, rather than a wider window, and then to attempt to achieve that target goal with the greatest possible precision, just as if one were shooting at any other target.

Individualized drug therapy therefore begins by setting such specific individualized target goals. Without specific target goals, there can be no individualized precise drug therapy. The task of the clinician is to select, and then to hit, the desired target goal as precisely as possible. As the initial regimen is given, the clinical task is to observe the patient's response, and to reevaluate whether the target goal was hit precisely or not, was correctly chosen or not, or if it should be changed and a new dosage regimen developed.

THE NEED FOR MODELS

Pharmacokinetic and dynamic models provide the means to store past experience with the behavior of drugs, and the tool to apply that past experience to the care of future patients. This experience is usually stored in the form of a population pharmacokinetic model which is used as the Bayesian prior to design the initial regimen for the next patient who appears to belong to that population. The dosage regimen to achieve the target goal is computed and given. The patient is then monitored both clinically and by measuring serum concentrations. The serum concentrations are used not only to note if they are within a therapeutic range, but also to make a specific model of the behavior of the drug in that individual patient. One can see what the probable serum concentrations were at all other times when they where not measured. One can also see the computed concentrations of drugs in a peripheral nonserum compartment or in various effect compartments. These cannot be seen or inferred at all without such models. By comparing the clinical behavior of the patient with the behavior of the patient's model, one can evaluate the patient's clinical sensitivity to the drug, and can adjust the target goal appropriately. For digoxin, for example, the inotropic effect of the drug correlates best with the behavior of the drug in the peripheral compartment rather than with the serum concentrations. The excellent model made by Reuning and colleagues for digoxin [1] has been highly useful clinically [2].

CURRENT BAYESIAN INDIVIDUALIZATION OF DRUG DOSAGE REGIMENS

The Maximum Aposteriori Probability (MAP) Bayesian approach to individualization of drug dosage regimens was introduced to the pharmacokinetic community by Sheiner et al. [3]. In this approach, parametric population models are used as the Bayesian priors. The credibility of these population models (their parameter variances) is then evaluated in relationship to those of the measured serum concentrations as they are obtained. The contribution of these two types of data and their variances to the MAP Bayesian posterior individualized patient model is shown in the objective function used, as shown below (1).

$$\sum \frac{(Cobs - C\,mod)^2}{Var\,(Cobs)} + \sum \frac{(Ppop - Pmod)^2}{Var\,(Ppop)} \qquad (1)$$

where Cobs is the collection of observed serum concentrations, Var (Cobs) is the collection of their respective variances, and Cmod is the model estimate of each serum concentration at the time it was obtained. Similarly, Ppop is the collection of the various population model parameter values, Var (Ppop) is the collection of their respective variances, and Pmod is the collection of the Bayesian posterior model parameter values. Each data point is given a weight according to its Fisher information, the reciprocal of its variance. Population models in which there is greater diversity, and therefore greater variance, contribute less to the individualized model than do more uniform models having smaller variances. Similarly, a precise assay will draw the fitting procedure more closely to the observed concentrations, and a less precise assay will do the opposite. The more serum data are obtained, the more that information dominates the determination of the MAP Bayesian posterior parameter values (Pmod) in the patient's individualized pharmacokinetic model.

Having made the patient's individualized model, one then uses it to reconstruct the past behavior of the drug in the patient during his therapy to date. One can examine a plot of the behavior of this model over the duration of the past therapy. One can thus evaluate the clinical sensitivity of the patient to the drug, by looking at the patient clinically and comparing the patient's clinical behavior with that of the patient's individualized

pharmacokinetic model. In that way, one can evaluate whether the initial target goal was well chosen or not. One can choose a different goal if needed, and once again one can compute the dosage regimen to achieve it. In this way, the model can be individualized and dosage can continue to be adjusted to the patient's body weight, renal function, and available serum concentrations, for example, to achieve the desired target goal, usually with increasing precision.

CRITIQUE OF THE MAP BAYESIAN APPROACH

The weakness of the MAP Bayesian procedure is that the models it uses have only single point estimates of the various pharmacokinetic parameters. Because of that, there is only one version of either the individualized model, or of the population model itself. The regimen developed to achieve the target goal is simply assumed to do so exactly.

THE SEPARATION PRINCIPLE

The separation or heuristic certainty equivalence principle [4] states that whenever control of a system is separated first, into obtaining single point parameter estimates for the model, and second, of using those single point estimates to control the system, the task is often achieved in a suboptimal manner. This is a significant problem with MAP Bayesian fitting and dosage design. The way around this problem is by incorporating improved nonparametric approaches to population pharmacokinetic modeling, and in using them specifically to design maximally precise dosage regimens.

USE OF POPULATION MODELS IN CLINICAL THERAPEUTICS

When a parametric population model is used as the Bayesian prior to design an initial dosage regimen for the next patient one encounters, one usually has only a single estimated value for each parameter. Because of this, only one prediction of future concentrations can be made. The dosage regimen is simply assumed to achieve the target goal exactly, as shown in Figure 2.

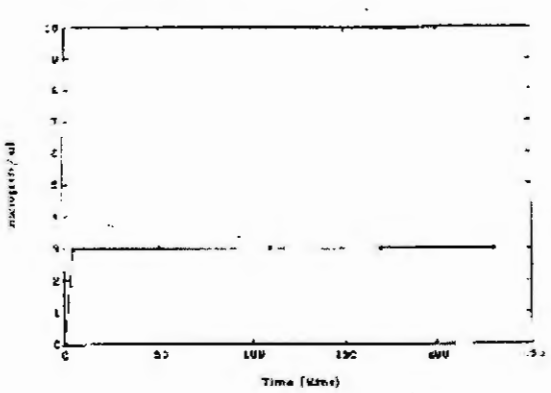


Figure 2. Using lidocaine population mean parameter values, an infusion regimen designed to achieve and maintain a target goal of 3 ug/ml does so exactly when the patient, as here, has exactly the mean population parameter values.

Figure 2 shows the results of an infusion regimen of lidocaine, based on the mean population parameter values for that drug, which was designed to achieve and maintain a target serum concentration of 3 ug/ml. As shown, this regimen, based on the single mean population parameter values, hits the target exactly, but only when the patient has parameter values which are exactly the population mean values.

However, as shown in Figure 3, when the regimen used in Figure 2 was given to the combination of the actual 81 diverse nonparametric population support points from which the mean values were obtained, an extremely wide distribution of predicted serum concentrations was seen, due to the diversity in the nonparametric population support points from which the mean parameter values were obtained. The predicted serum concentrations actually covered much more than the usual therapeutic range of 2 to 6 ug/ml.

In contrast, if one has a nonparametric population model [5-8], with its multiple sets of model parameter values (81 in this case), one can make multiple predictions, instead of only one, forward into the future from any candidate dosage regimen which is "given" to all the models in the population discrete joint

density. The richer and more likely population parameter joint density reflects better the actual diversity among the subjects studied in the past population. Based on these multiple models in the population (the discrete joint density), one can compute

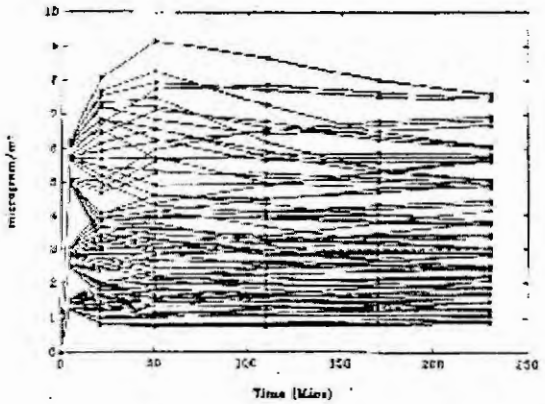


Figure 3. Result when the above lidocaine infusion based on population mean parameter values is given to the 81 diverse support points from which the population mean values were obtained. Great diversity in the predicted responses is seen.

the weighted squared error with which any candidate regimen is predicted to fail to achieve the desired target goal at a target time. Other regimens can then be considered, and the optimal regimen can be found which is specifically designed to achieve the desired target goal with the least weighted squared error [9-11].

This approach, using the multiple models of the patient provided by the nonparametric population model, avoids much of the limitations of the separation principle. This is the real strength of the combination of nonparametric population models coupled with "multiple model" dosage design [9-11].

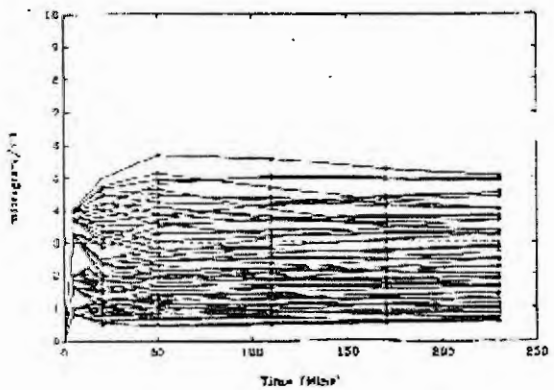


Figure 4. Predicted response of the 81 support points (models) when the regimen obtained by multiple model dosage design is given. The target is achieved with visibly greater, and optimal, precision.

As shown in Figure 4, the multiple model (MM) dosage regimen, based on the same nonparametric population model with its 81 support points, obtained a much more precise achievement of the target goal, because it was specially designed to do so. The error in the achievement of the therapeutic target goal is much less, and the dispersion of predicted serum concentrations about the target goal is much less. Other cost functions can also be used [13,14].

OBTAINING MULTIPLE MODEL BAYESIAN POSTERIOR JOINT DENSITIES

With the MAP Bayesian approach to posterior parameter values, the single most likely value for each parameter is obtained when they altogether minimize the objective function shown in equation (1). In contrast, the MM Bayesian approach, using the nonparametric joint densities, preserves the multiple sets of population parameter values, but specifically recomputes their Bayesian posterior probability, based upon the serum concentrations obtained. Those combinations of parameter values that predicted the measured concentrations

well become more probable. Those that predicted them less well become less so. In this way, the probabilities of all the nonparametric population support points become revised, using Bayes' theorem [10-11]. A smaller number of significant points, or perhaps even only one, is usually obtained. When the regimen for the next cycle is developed, these revised models, containing their revised MM Bayesian posterior probabilities, are used to develop it. The regimen is again specifically designed to achieve the desired target goal with maximum precision (minimum weighted squared error).

OTHER BAYESIAN APPROACHES

Three other Bayesian approaches have been used by us to incorporate feedback from measured serum concentration data. The first is the sequential MAP Bayesian approach, in which the MAP posterior parameter values are sequentially updated after each serum concentration data point is obtained. This procedure improves the tracking of the behavior of the drug through each data set. However, at the end of each full feedback cycle, (after each new full cluster of data points), at the time the next regimen is to be developed, this method has learned no more with respect to developing the next new dosage regimen, than if it had fitted all the data together at once, even though it tracks the changing MAP Bayesian parameter values better sequentially.

The second approach is the sequential MM Bayesian one [9-11]. Here the MM Bayesian posterior joint density is also sequentially updated after each data point. Still, at the end of each feedback cycle, this procedure similarly has learned no more with respect to developing the next dosage regimen than if all the data in that cluster were fitted simultaneously. The procedure is still looking for a hypothetical single model (support point, set of parameter values) which best describes all the data. When this fails to be the case, combinations of support points are found which fit best. Still, the procedure is looking for a fixed and unchanging single model, or combination of models, which best fit the data, even though the posteriors are fitted sequentially.

A third approach is the interacting multiple model (IMM) approach [12]. This method permits the true patient being sought for actually to jump from one model or support point to another during the sequential Bayesian analysis. Because of this the IMM method, originally designed to track missiles and aircraft taking evasive action, permits detection of changing pharmacokinetic parameter densities during the sequential analysis procedure. It thus provides an improved method to track the changing parameter densities and behavior of a patient during the evolution of his clinical therapy. For example, it permits an improved ability to detect and to quantify changes in the volume of distribution of aminoglycoside drugs during changes in a patient's clinical status which are not captured by the use of conventional clinical descriptors. Using carefully simulated models in which the true parameter values changed during the data collection, the integrated total error in tracking a simulated patient was very similar with the sequential MAP and sequential MM Bayesian procedures. However, the integrated total error of the sequential IMM prodecure was only about one half that of the other two [12].

CLINICAL APPLICATIONS

Nonparametric population parameter joint densities, MM dosage design and IMM Bayesian posterior joint densities appear to offer significant improvements in the ability to track the behavior of drugs in patients during their care, especially when the patients are unstable and have changing parameeer values. These approaches also develop dosage regimens which are specifically designed to achieve target goals with maximum precision. These methods make optimal use of all information contained in the past population data, coupled with whatever current data of feedback may be available about a particular patient up to that point, to develop that patient's most precise dosage regimen. A clinical version of this software, which runs on PC's in Windows, is now in development.

Acknowledgements: Supported by NIH grants LM 05401 and RR 11526.

References.

1. Reuning R, Sams R, and Notari R: Role of Pharmacokinetics in Drug Dosage Adjustment: 1. Pharmacologic effects. Kinetics, and apparent volume of distribution of Digoxin. J. Clin. Pharmacol. 13: 127-141, 1973.
2. Jelliffe R, Schumitzky A, Van Guilder M, et al.: Individualizing Drug Dosage Regimens: Roles of Population Pharmacokinetic Models, Bayesian Fitting, and Adaptive Control. Ther. Drug Monit., 15: 380-393, 1993.
3. Sheiner L, Beal S, Rosenberg B et al.: Forecasting Individual Pharmacokinetics. Clin. Pharmacol. Therap. 26: 294-305, 1979.

4. Bertsekas D: Dynamic Programming: Deterministic and Stochastic Models. Prentice-Hall, Englewood NJ, 1987, pp. 144-146.
5. Lindsay B: The Geometry of Mixture Likelihoods: a General Theory. Ann. Statist. 11: 86-94, 1983.
6. Mallet A: A Maximum Likelihood Estimation Method for Random Coefficient Regression Models. Biometrika 73: 645-656, 1986.
7. Schumitzky A: Nonparametric EM Algorithms for Estimating Prior Distributions. App. Math. Comput. 45: 143-157, 1991.
8. Schumitzky A: The Nonparametric Maximum Likelihood Approach to Pharmacokinetic Population Analysis. Proceedings of the 1993 Western Simulation Multiconference: Simulation for Health Care. San Diego Society for Computer Simulation, pp. 95-100, 1993.
9. Bayard D, Milman M and Schumitzky A: Design of Dosage Regimens: a Multiple Model Stochastic Control Approach. Int. J. Biomed. Comput. 36: 103-115, 1994.
10. Bayard D, Jelliffe R, Schumitzky A, Milman M, and Van Guilder M: Precision Drug Dosage Regimens using Multiple Model Adaptiva Control: Theory and Application to Simulated Vancomycin Therapy. in Selected Topics in Mathematical Physics, Professor R. Vasudevan Memorial Issue, ed. By Sridhar R, Srinavasa Rao K, and Lakshminarayanan V, Allied Publishers Ltd., Madras, India, 1995, pp. 407-426.
11. Jelliffe R, Schumitzky A, Bayard D, MilmanM, Van Guilder M, Wang X, Jiang F, Barbaut X, and Maire P: Model-Based, Goal-Oriented, Individualized Drug Therapy: Linkage of Population Modelling, New "Multiple Model" Dosage Design, Bayesian Feedback, and Individualized Target Goals. Clin. Pharmacokinet. 34: 57-77, 1998.
12. Bayard D and Jelliffe R: Bayesian Estimation of Posterior Densities for Pharmacokinetic Models having Changing Parameter Values. To be presented at the Tenth Annual International Conference on Health Sciences Simulation, San Diego CA, January 23-27, 2000.
13. Tarıght N, Mentre F, Mallet A, and Jouvent R: Nonparametric Estimation of Population Characteristics of the Kinetics of Lithium from Observational and Experimental Data: Individualization of Chronic Dosing Regimen using a new Bayesian Approach. Ther. Drug Monit. 16: 258-269, 1994.
14. Jerling M: Population Kinetics of Antidepressant and Neuroleptic Drugs: Studies on Therapeutic Drug Monitoring Data to Evaluate Kinetic Variability, Drug Interactions, Nonlinear Kinetics, and the Influence of Genetic Factors. Ph.D. Thesis. Stockholm: Karolinska Institute at Huddinge University Hospital, pp. 28-29, 1995.

EXPERT KNOWLEDGE INCLUSION IN PHARMACOKINETICAL MODELS

A. Belič, R. Karba, *I. Grabnar, *A. Mrhar and B. Potočnik
Faculty of Electrical Engineering
Tržaška 25, 1000 Ljubljana
*Faculty of Pharmacy
Aškerčeva 7, 1000 Ljubljana
Slovenia

Abstract. Measurements in biological tissues and liquids are often problematic. Ethical, physical, and financial limitations do not allow many samples to be taken from living organisms and thus precious information on system dynamics is not available. At the same time accuracy and stabillity of measurement methods is rather poor in compare to technical sciences. To eliminate noise from the data and to reconstruct the time course of the system dynamics, splines can be used. The time course is determined from expert knowledge and sampled data by using weighted least square method for spline fitting. Weights of data points are determined by the expert. The time course described by splines, instead of raw measurements, is now used in modelling procedures. In this work the approach was tested on two substances; histamine and paracetamol. Such procedure showed substantial advatage in the process of determination of model parameters.

Introduction

Measurements in biological tissues and liquids are often problematic. Ethical, physical, and financial limitations do not allow many samples to be taken from living organisms and thus precious information on system dynamics is not available. At the same time accuracy and stabillity of measurement methods is rather poor in compare to technical sciences. Pharmacokinetics (PK) [10, 6] is faced with the same problem. Thus analysis of such noisy and sparse data may lead to improper conclusions about system mechanisms and provides significant problems to modelling. On the other hand the expert knowledge on system dynamics exists and cannot be fully exploited. To eliminate noise from the data and to reconstruct the time course of the system dynamics, splines can be used. The time course is determined from expert knowledge and sampled data by using weighted least square method for spline fitting. Weights of data points are determined by the expert. The time course described by splines, instead of raw measurements, is now used in modelling procedures [8, 3]. Model structure is based mainly on expert knowledge and preliminary biological experiments, however, model parameters are defined by fitting model responses to the measurements approximated by splines. In this work the approach was tested on two substances; histamine and paracetamol.

Splines

Splines are piecewise polynomial curves [9, 2].

$$\hat{f}(t) = \begin{cases} p_1(t) & ;0 \leq t < t_1 \\ p_2(t) & ;t_1 \leq t < t_2 \\ \vdots & \\ p_n(t) & ;t_{n-1} \leq t \leq t_n \end{cases} \tag{1}$$

At the transition point from one polynomial to the next the following conditions must be conserved:

$$
\begin{aligned}
p_1(0) &= f(0) && (2)\\
p_{i-1}(t_{i-1}) &= p_i(t_{i-1}) && (3)\\
p_{i-1}^{(1)}(t_{i-1}) &= p_i^{(1)}(t_{i-1}) && (4)\\
p_{i-1}^{(2)}(t_{i-1}) &= p_i^{(2)}(t_{i-1}) && (5)\\
&\vdots && (6)\\
p_{i-1}^{(n-1)}(t_{i-1}) &= p_i^{(n-1)}(t_{i-1}) && (7)\\
p_n(t_n) &= f(t_n) && (8)
\end{aligned}
$$

In most cases third order polynomials are used, since they imply continuous curve up to its second derivative. That is enough for mechanical and mass transport systems. Since PK systems are mass transport sistem, third order polynomials are sufficient. The use of polynomials instead of linear interpolation provides more natural, smooth time course. On the other hand exponential curves prejudicate model structure too rigidly and are, therefore, not suitable for this task.

Data filtering by splines

Since PK data is usualy nonequidistantly sampled classical statistical denoising methods can not be applied. Therefore splines were proposed as one of possible approaches. Nonequidstant and noisy data is approximated by spline using weighted least squares to determine third order polynomials best fitting the data. Since system dynamics is poorly described by such a sparse data, simple least squares could produce inacceptable shape of the spline, therefore expert knowledge is used to determine weights associated with each sample (see figure 1).

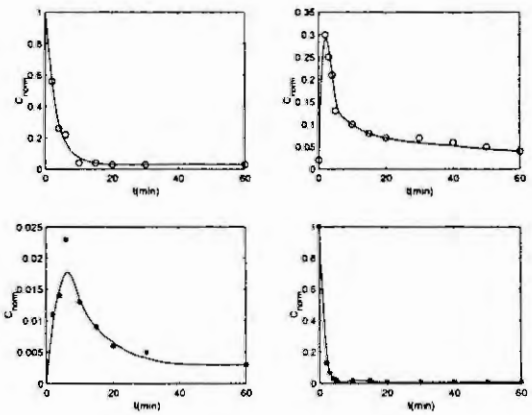


Figure 1: Histamine data filtered by splines. o - measured data, line - spline.

Parameter estimation of the model

Parameter estimation is often made by adaptive model scheme where model parameters are varied by optimisation method to obtain best possible fitting of model response and measured data. Since PK models that include physiological information often tend to nonidentifiability [7, 5] optimisation method can easy find a solution that is good considering only a measure of fit and is totally unacceptable considering the shape of the model response. Oscillations in a curve usually give better measure of fit since the data is sparse, however, expert, knowing the dynamics of the system, would not accept such a solution

Experiments

The proposed procedure was tested on two substances, where difficulties finding an acceptable set of model parameters occured. The oscillations in model response mainly provided better fitting to measured samples than smooth curve, however, the oscillations were physiologically unacceptable.

Histamine [1]

Optimal parameter set, considering best measure of fit gave oscillatory model response (see Figure 2 a)) when raw data was used as a reference curve. Therefore reference curve was replaced by splines (see

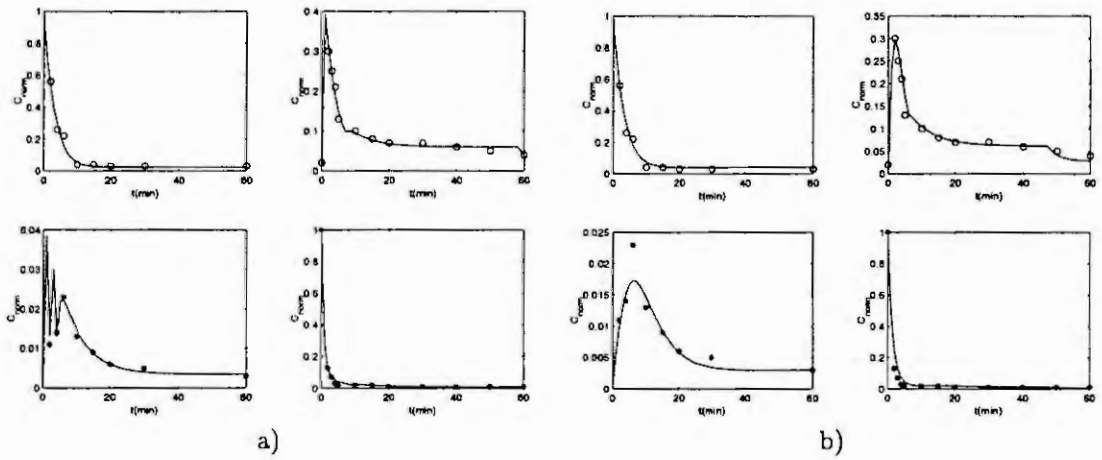


Figure 2: The effect of splines on curve fitting. Figure a) - unacceptable fit (raw data as reference). Figure b) - acceptable fit (spline filtered data as reference curve)

Figure 1). Optimisation procedure was now able to find acceptable solutions (see figure 2 b))

Paracetamol [4]

The case of nonidentifiable model and very noisy data is treated. However, the problem was that concentrations in plasma were measured for too short period of time causing substantial problems for parameter estimation process. After the last measured point, simulated curve droped to zero (see Figure 4 a)), since there was no information available after that point. However, cumulative amount of excreted paracetamol was also monitored in urine and according to those measurements and known phyisiology, we were able to reconstruct the missing part of plasma concentration data (see Figure 3). Replacing measured data by spline approximations again proved to be successful since the problems were

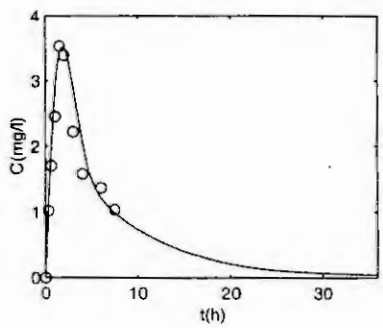


Figure 3: Paracetamol data filtered by splines. o - measured data, line - spline.

substantially reduced (see Figure 4 b).

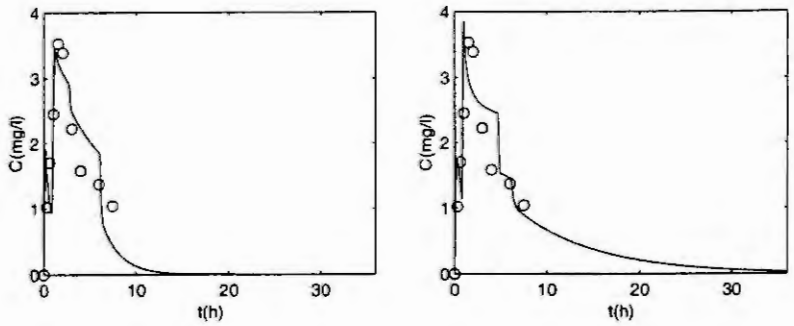


Figure 4: The effect of splines on curve fitting. Figure a) - unacceptable fit (raw data as reference). Figure b) - acceptable fit (spline filtered data as reference curve)

Summary

We can summarize as follows:

- Measurements in bimedicine produce sparse noisy data that is usually nonequidistantly sampled.
- In a lot of cases expert knowledge exists, however, it is never included in modelling process.
- Estimation of model parameters is problematic when dealing both with nonidentifiable models and sparse noisy data.
- Splines can successfully filter sparse nonequidistantly sampled data with the help of expert knowledge.
- Approximated measurements by splines speed up model parameter estimation process and assure acceptable model dynamics.

References

[1] Belič, A., Grabnar, I., Karba, R., Mrhar, A., Irman-Florjanc, T. Primožič, S.: Interdependence of Histamine and Methylhistamine Kinetics: Modelling and Simulation Approach. Computers in Biology and Medicine, 29(6):361–375, 1999.

[2] Boor, C. D.: A Practical Guide to Splines. Springer Verlag, New York, 1978.

[3] Cellier, F. E.: Continuous System Modelling. Springer-Verlag, New York, 1991.

[4] Chicco, D., Grabnar, I., Škerjanec, A., Vojnovic, D., Maurich, V., Realdon, N., Ragazzi, E., Belič, A., Karba, R. Mrhar, A.: Correlation of in vitro and in vivo paracetamol availability from layered excipient suppositories. International Journal of Pharmaceutics, 189(2):147–160, 1999.

[5] Eykhoff, P.: System Identification. John Wiley & Sons, London, 1974.

[6] Gibaldi, M. D. Perier: Pharmacokinetics. Marcel Dekker, New York, 1982.

[7] Godfrey, K.: Compartmental Models and Their Application. Academic Press, London, 1983.

[8] Matko, D., Karba, R. Zupančič, B.: Simulation and Modelling of Continuous Systems: A Case Study Approach. Prentice Hall, New York, 1992.

[9] Schumaker, L. L.: Spline Functions: Basic Theory. Krieger Publishing Company, Malabar, Florida, USA, 1993.

[10] Wagner, J. G.: Pharmacokinetics for the Pharmaceutical Scientist. Technomic Publishing inc., Lancaster, 1993.

THE ROLE OF GENETIC ALGORITHMS IN PHARMACOKINETIC - PHARMACODYNAMIC MODELLING AND EVALUATION

A. Belič, R. Karba, *I. Grabnar, *A. Mrhar and D. Andrejič
Faculty of Electrical Engineering
Tržaška 25, 1000 Ljubljana
*Faculty of Pharmacy
Aškerčeva 7, 1000 Ljubljana
Slovenia

Abstract. Pharmacokinetic-Pharmacodynamic (PK-PD) models are often nonidentifiable, especially when their structure is derived from physiological characteristics of the system rather than from measurements only. However, model parameter have to be estimated from the measurements for each individual, when intersubject variability is considerable. The choice of optimisation method is therefore of great importance since the criterium function in the case of nonidentifiable model exhibits a great number of extremes. Conventional optimisation methods are not very effective, since they start in one point of parameter space and move towards a criteria function extreme. Genetic algorithms, however, start with a whole population of starting points randomly spread through the parameter space and by applying laws of genetics and selection they search for the best parameter set representing one subject in the population. Parameter space is therefore better explored than with conventional methods. Parallel genetic algorithms outperform simple genetic algoritms in cases of searching. Both genetic algorithms were tested in cases of histamine and paracetamol. Parallel GA requires smaller number of runs to obtain reliable information on model parameters.

Introduction

Pharmacokinetic-Pharmacodynamic (PK-PD) models [11, 7, 6] are often nonidentifiable [7, 4], especially when their structure is derived from physiological characteristics of the system rather than from measurements alone. However, model parameters have to be estimated from the measurements for each individual, since intersubject variability is considerable. To estimate model parameter values, scheme with adaptive model is used, where model parameter values are varied to achieve best possible fit of model response to the measured curve (reference curve). The choice of optimisation method is of great importance since the criterum function in the case of nonidentifiable model exhibits a great number of extremes. Some optimal sets of parameters are better and some worse, therefore a thourough search through entire parameter space must be undertaken. Conventional optimisation methods are not very effective, since they start in one point of problem space and move towards a criteria function extreme. Genetic algorithms (GA) [8, 5], however, start with a whole population of starting points randomly spread through parameter space and by applying laws of genetics and natural selection they search for the best parameter set representing one subject in the population. Parameter space is therefore better explored than with conventional methods.

Genetic algorithms

Basic idea of GA is to search for optimal solution by applying natural laws of genetics and evolution. Different GAs have been proposed and tested in the literature, each being optimal for given problem. However, there was no universal solution for all optimisation problems and very likely never will be. In this work we have tested two GAs; simple GA and parallel GA [9]. It has been known [9] that larger populations of problem solutions are not allways better in problem space exploration. It is suspected

that best genetic material e.g. parameter set, prevails too slowly over less succesful subjects than in smaller populations. Therefore, the convergence of algorithm is much slower and results are usually furhter away from true optimum. On the other hand, parallel algorithm divides the whole population into smaller subpopulations. That enables faster convergence of the solution in subpopulations and at the same time algorithm explores the problem space better since each subpopulation usually distributes over different part of problem space. However from time to time migrations between subpopulations are alowed. This model is much closer to natural evolution of species and also works considerably better in problem space exploration.

Estimation of model parameters

Final stage in each modelling cycle [10, 2] is model parameter estimation. When model structure is derived from physiological characteristics it is usually nonidentifiable and unique solution of the problem does not exist. However, it is sufficient to find an acceptable solution in the set of all solutions. That, however, requires a method that is not very exact but can explore large parameter spaces in relatively short time. GAs represent a successful compromise between stohastic and deterministic search and optimisation algorithms. By balancing stohastic and determimistic component of the algorithm, convergence speed can be set to achieve the desired effect. When searching in large parameter spaces and when exact solution is not very important, stohastic component must prevail over deterministic component. That is just what the modelling in PK-PD requires.

Experiments

Two algorithms were tested; parallel and simple. Both were tested on histamine [1] and paracetamol [3] model. Histamine is endogene amine present in humans, animals, and platnts. It plays an important role in alergic reactions. Paracetamol is used as an analgesic and antipiretic drug.

Paracetamol

Two tests were made on paracetamol model. First both algoritms were tested when they were set to be equaly time consuming. Simple algoritm namely had a population of 30 subjects and ran for 1000 generations, parallel algorithm had 5 subpopulations of 30 subjects and ran only 200 generations all other settings were the same. Each optimisation run was repeated 21 times. At the same time the efficiency of both algorithms with the same uumber of generations was tested. Criteria for effectiveness was final value of criteria functions and visual acceptance of model response, since criteria functions in most cases can not completley describe desired goals. Results are presented in table 1. As can be seen in Table 1, when nubmer of generations for simple algorithm was set to 1000 it was able to get the best result (run 18), however, its worst result was worse than the worst result of parallel algorithm. The consistency of results of parallel algorithm is also much better than of simple algoritm. When number of generations was the same for both algorithms parallel algoritem clearly outmatched simple algorithm.

Histamine

Both algorithms were also tested on histamine case. The results were surprising, since visually all results of parallel algorithm were unacceptable. However, the problem was in criteria function, since oscillatory model responses had lower value of citeria functions than nonoscillatory. Again parallel algorithm was more consistent in results. In Figure 1 most frequent solutions of both algorithms are shown.

Summary

We can summarize as follows:

- PK-PD models are often nonidentifiable, therefore it is difficult to determine acceptable sets of model parameters.

	Simple GA 1000 gen.		Parallel GA 200 gen.		Simple GA 200 gen.	
run	crit.f.	visual q.	crit.f.	visual q.	crit.f.	visual q.
1	0.2372	$\neq$	0.1703	$=$	0.3274	$\neq$
2	0.1771	$\neq$	0.1032	$\approx$	0.2537	$\neq$
3	0.2355	$\approx$	0.1134	$\approx$	0.1550	$\approx$
4	0.2061	$\approx$	0.1737	$\approx$	0.3389	$\neq$
5	0.1891	$\approx$	0.1535	$\approx$	0.2849	$\neq$
6	0.1993	$\neq$	0.1122	$\approx$	0.2851	$\neq$
7	0.1716	$\approx$	0.2336	$\approx$	0.1564	$\approx$
8	0.1534	$=$	0.1501	$=$	0.1971	$\approx$
9	0.0861	$=$	0.1407	$=$	0.2392	$\neq$
10	0.2033	$\neq$	0.1130	$=$	0.3073	$\neq$
11	0.1890	$\approx$	0.1392	$=$	0.3306	$\neq$
12	0.2834	$\neq$	0.1549	$=$	0.2604	$\neq$
13	0.1721	$\approx$	0.1517	$=$	0.3689	$\neq$
14	0.1734	$\approx$	0.1949	$=$	0.2227	$\neq$
15	0.2616	$\neq$	0.1436	$\approx$	0.2817	$\neq$
16	0.1358	$\approx$	0.1763	$\approx$	0.2887	$\neq$
17	0.2833	$\neq$	0.1430	$=$	0.1956	$\approx$
18	0.0803	$=$	0.1731	$\approx$	0.2828	$\neq$
19	0.2781	$\neq$	0.1725	$\approx$	0.3070	$\neq$
20	0.0967	$=$	0.1172	$=$	0.2514	$\neq$
21	0.0660	$=$	0.1345	$=$	—	—

Table 1: Results of the test. In column of visual quality the symbols have the following meaning: $\neq$ - bad result, $\approx$ - acceptable result, $=$ - good result.

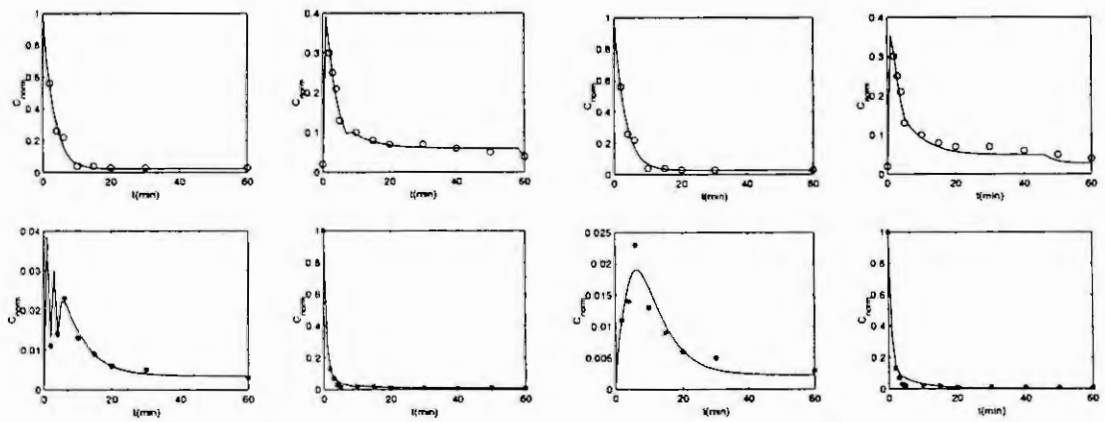


Figure 1: Curve fitting with GA. Figure a) - most frequent solution of parallel algorithm. Figure b) - most frequent solution of simple GA.

- Since the number of model parameters is great, a parameter space is of high dimension, therefore, an algorithm to search the space effectivelly but not too exactly is needed.
- GAs are a good choice in estimation of nonidentifiable model's parameters.
- Parallel GA is more efficient in exploring the high dimensional spaces and altough it is more time consuming than simple GA, it provides better consistency and therefore higher reliability of the results.

- To obtain a reliable information on model parameter values, optimisation runs must be repeated several times regardless of the optimisation method. However, parallel GA requires smaller number of runs, since it is able to explore larger portions of parameter space in one run.

References

[1] Belič, A., Grabnar I., Karba R., Mrhar A., Irman-Florjanc T. Primožič S.: Interdependence of Histamine and Methylhistamine Kinetics: Modelling and Simulation Approach. Computers in Biology and Medicine, 29(6):361–375, 1999.

[2] Cellier, F. E.: Continuous System Modelling. Springer-Verlag, New York, 1991.

[3] Chicco, D., Grabnar I., Škerjanec A., Vojnovic D., Maurich V., Realdon N., Ragazzi E., Belič A., Karba R. Mrhar A.: Correlation of in vitro and in vivo paracetamol availability from layered excipient suppositories. International Journal of Pharmaceutics, 189(2):147–160, 1999.

[4] Eykhoff, P.: System Identification. John Wiley & Sons, London, 1974.

[5] Gen, M. Cheng R.: Genetic Algorithms & Engineering Design. John Wiley & sons, inc., New York, 1997.

[6] Gibaldi, M. Perier D.: Pharmacokinetics. Marcel Dekker, New York, 1982.

[7] Godfrey, K.: Compartmental Models and Their Application. Academic Press, London, 1983.

[8] Goldberg, D. E.: Genetic Algorithms in Search, Optimization, and Machine Learning. Addison Wesley Publishing Company, New York, 1989.

[9] Levine, D.: A Parallel Genetic Algorithm for the set Partitioning Problem. Doktorska disertacija ANL 94/23, Argonne National Laboratory, Argonne, IL, USA, May 1994.

[10] Matko, D., Karba R. Zupančič B.: Simulation and Modelling of Continuous Systems: A Case Study Approach. Prentice Hall, New York, 1992.

[11] Wagner, J. G.: Pharmacokinetics for the Pharmaceutical Scientist. Technomic Publishing inc., Lancaster, 1993.

SOFT COMPUTING METHODOLOGY IN MODELING AND SIMULATION IN MEDICINE

Dietmar P.F. Möller
*University of Hamburg, Department Computer Science, Chair Computer Engineering
& McLeod Institute of Simulation Sciences, German Chapter at University of Hamburg
Vogt-Kölln-Str. 30, D-22527 Hamburg, Germany
Tel.: ++49-40-42883-2438. Fax: ++49-40-42883-2552,
Email: dietmar.moeller@informatik.uni-hamburg.de

Abstract. This paper presents the application of soft computing methodology in modeling and simulation in medicine. Soft computing methodology is introduced as part of sensitive neural networks and fuzzy-classificators. While sensitised neural nets enable physicians the conditioning of patient specific neural classifiers, physicians are able using specific linguistic IF-THEN rules to create appropriate fuzzy sets for modeling and simulation purposes. Due to that, physicians will be empowered handling this new classifiers in situ, phased to their medical equipment and/or patient specific parameters, describing individual orthological as well as pathological states.

1. Theory of fuzzy logic for a medical supporting system

The features of fuzzy logic can be summarised as follows: Approximate expressions are quantified by using membership functions representing all the possible input values, and rules are processed in a parallel and cohesive manner. So fuzzy-logic, based on the fuzzy-set theory, allows the physician to describe a medical situation of a patient linguistically, which means in words, instead of terms of mathematics, which can be represented as the adequate way of describing medical reasoning [1].

Assuming that input variables of the medical situation under test are x_1 and x_2 and the output value is y, the IF-THEN-rules could be given in the following way:

IF x_1 is A_1 AND x_2 is B_1 THEN $y = C_1$, ..., IF x_1 is A_n AND x_2 is B_n THEN $y = C_n$

In this example the input values A or B_1 can be indefinite values or relative terms such as *large, small, big, high, fast, slow*, etc.; they are the input variable of the respective fuzzy-set. The output value C can be a relative value relating the output variable to the fuzzy set. A membership function represents each of the fuzzy sets and acts as a transformation between crisp and fuzzy values.

2. The theory of the sensitization of neural nets

The idea to condition a neural net first by well defined easy distinguishable data sets and then to deepen and to enlarge the stored information by sensitization we learned from cognitive psychology in the context of the chunking problem. Chunking is more or less the adaptation of a new fact or a so far unknown situation with the help of knowledge facts or models. Only out of old facts or acting strategies man can develop new strategies for understanding of so far unknown. Transforming this model to the handling of neural nets means, that first a net has to learn a basis concept. To prevent that the net includes sensor typical output ranges in its classification behaviour and therefore looks for sensors with a high output, it is necessary to normalise the input data set, by using a appropriate-processing [3]. To handle only the changes in the global state of the patient it is favourable to use a preprocessing step which turns out a gradient vector which is calculated by the difference between the data of a patient as requested when no pathological situation is present minus the present actual data.

Once the neural classifier can separate all trained states which represent a basic concept in a sufficient way the weaker evolutionary states of the different pathological states should be trained. The definition of the basic concept or as we called: "the worst cases of a system's/patient's behaviour" is simply to be defined as the problem oriented data set representing a classification state satisfactory.

By presenting the weaker states after the basic concept is settled it will be ensured that the net will be forced to change his classification structure slightly out of it's former structure, without destroying the older structure. We say the net will be sensitized. Especially when a backpropagation network is used the learning rules force the net to sensitise its structure in that way that only the pathological state representing structure is modified, as the classification results have to be the same over the whole sensitization period. If weaker and weaker states will be presented successively the classification structure will change accordingly, until the similarity of the different evolutionary state representations will be so little, that the net can not be forced to change it's structure anymore.

Figure 2 a and 2b show a sensitised net work structure graphically. Since the classification potential is only changed locally, the net changes its classification behaviour not generally by learning the evolutionary data sets,

but shows the adaptive behaviour to them accordingly. This local change surely can lead to the unification of so far divided concepts, a fact which will open the door to a wide range of so far unknown or unnoticed inner connections of the data sets, representing the patient's states.

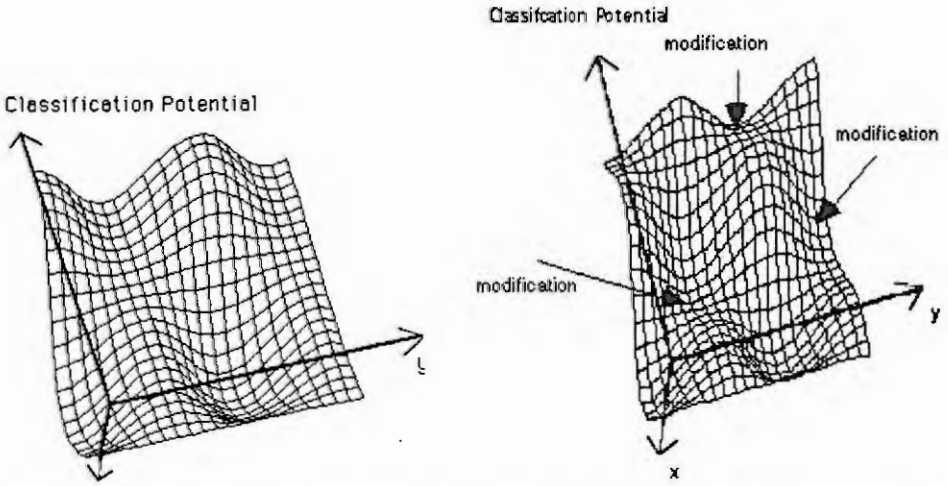


Fig. 2: classification-potential a) E before a sensitisation and b) E' after a sensitisation

3. Experimental results

The ECG indicates the heart action and function. Normally ECG is recorded in lead groups simultaneously, while the ECG leads are assumed to be projections of the electrical heart vector onto the equilateral so-called Einthoven triangle in the frontal plane only.

The normal ECG is composed of a P wave and a QRS complex and a T wave. The QRS complex represents actually three separated waves, the Q, the R and the S wave.

The main goal of applying fuzzy logic to ECG-analysis is the identification of abnormalities in the ECG, e.g.

- abnormalities in the QRS complex, which can be stated as coronary ischaemia or angina pectoris;
- abnormalities in the T wave, which can be stated as apex ischaemia, left bundle branch block, right bundle branch block, ventricular extrasystole, or otherwise.

Rhythmicity and its abnormalities, clinically called cardially arrhythmia's, are one of the major facts to be detected, in order to prevent myocardial infarct situations for patients with an elevated risk.

The medical knowledge for an ECG-analysis can be transformed in fuzzy-rules, which contain binary fuzzy relationships between antecedents and consequence. They take their membership-values within the interval [0,1] U {v} with v: no relationship, within the fuzzy model.

The complex fuzzy ECG model can be described as follows:

where the input variables characterise the following affiliations:

filter:	2nd derivation of measure
drold:	relation between R-R-interval distance and time between interval and previous peak
level:	actual measure
freq:	heart frequency
dQs:	relation between actual QRS complex width and previous QRS complex width

The output variables q, r, s characterise the expected values of Q, R, S, pause/asystolia and arrhythmia. The rule base for QRS detection contains 9 rules, three of them are shown below:

If 'filter' is P and 'level' is N and 'drold' is S then q = yes

If 'filter' is Z and 'level' is N and 'drold' is S then q =possible

If 'filter' is NB or 'level' is N then q = no

The rule base for arrhythmia detection is based on 12 rules, two of them are shown here to give an impression for the case pause/asystolia.

IF filter is Z AND drold is VB AND level is N THEN pause = yes

IF filter is NB AND drold is N AND level is PB THEN pause = no

The fuzzy EGC-analysis, based on a fuzzy model kernel and its simulation, have been realised on a PC using MS-Windows and a FUZZY-SHELL, which contain the used min- max-operators.

Fuzzy system for anaesthesia control

During routine clinical work, the anaesthesiologist is mainly interested in continuos measurement of several important quantities like blood pressure, expired carbon dioxide, inspired and expired oxygen concentration, anaesthesia gas concentration etc., in order to satisfy the patient's actual needs during anaesthesia, as well as the regulation of the depth of anaesthesia, which is the ultimate goal of quantitative anaesthesia.

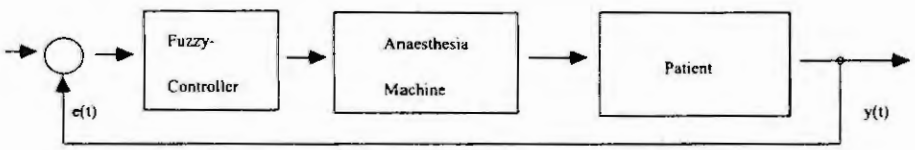
The regulation of the depth of the anaesthesia is predicated on the existence of a measurable pharmacokinetic and/or blood pressure signal as a correlate of depth of anaesthesia, as well as on the existence of a quantitative description of the influence of the different anaesthetic agents on this correlate.

It is well known from different reports, that today an appropriate correlate of the depth of anaesthesia has been found with the digital processing of the EEG. Because anaesthetic agents influence the power spectrum of the EEG in a characteristic manner as well as anaesthesia events. As an example each time the EEG crossed zero, a bolus of an anaesthesia agent was given by a syringe which was attached to a stepper motor. As the patient became anaesthetised the infusion rate slowed because the frequency of the EEG decreased. The median of the EEG-power spectrum can serve as the feedback signal within regulation loop of depth of anaesthesia [1]. The design of the regulation loop has to be guided by the knowledge of the anaesthetic agents used. But today a problem arises, that EEG is not yet a standardized measure during daily anaesthesia routine. Therefore anaesthesiologists today still use blood pressure as one of the most reliable signal for dosing anaesthetic agents.

Closed-loop regulation has been very successfully to the control of mean blood pressure by the regulation of sodium niroprusside infusion. Several regulator strategies have been used, including classical PID control, adaptive PID control and so called sophisticated selfrunner or adaptive control and model reference control [7].

A fuzzy logic based mean blood pressure regulation during anaesthesia is related between inflow concentration of the anaesthetic u(t) and the resulting mean blood pressure y(t). For the fuzzy controller design the error e(t), the change of error ce(t) and the integral of error ie(t) were used to generate the control variable y(t). The input variables u_i and the output variables Y are related as follows:

The block diagram of the regulation system for control of anaesthesia is given below:



Based on the linguistic rules shown, simulation studies have been done to find out a set of appropriate rules for stable system response behaviour. The first results obtained from our investigations indicate, that the fuzzy regulation of the depth of anaesthesia with mean blood pressure as the most reliable signal for dosing anaesthesia agents, can be an impact of the ultimate goals of quantitative anaesthesia as well as of quality assurance during anaesthesia.

Experimental results with sensitized neural nets

To create a supporting system which announce slightly occurring behaviour changes in the heart beat signal and support the physician by presenting him the probability of concepts regarding the early future of the beat behaviour a common back-propagation network with 124 input neurones 48 hidden neurones and 9 output neurones was used. The learning parameters were chosen as 0.8 for the momentum term and 0.1 for

The 4 output neurones represent the states of the patients heard activity (normal activity, arrhythmia (bigem), tachycardia ventricularis, pace maker).

The use of the frequency representation of the heart activity was chosen to neglect phase correlation between the learning sets and the classification sets and therefore to do the classification without a trigger unit.

In Figure 3a and 3b the time curves for the first and fourth heart signal are given.

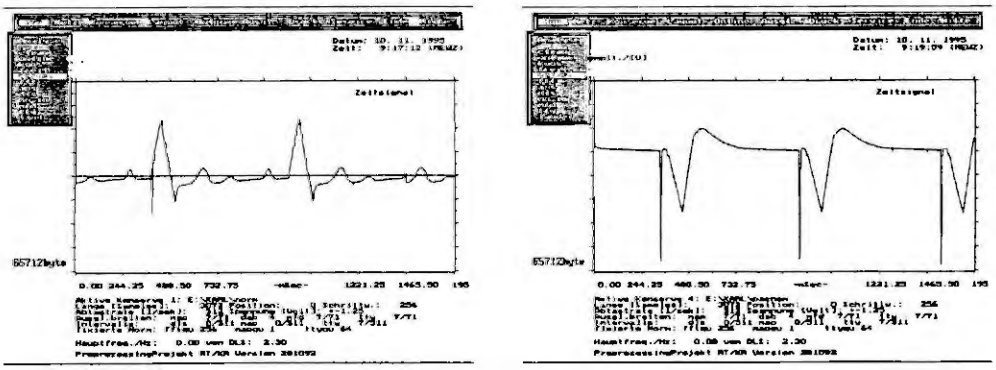


Fig. 3a: Normal heart beat signal, 3b: Pace maker triggered heart signal

Figure 3 gives an example of the interface of a sensitized network supporting system. While the first horizontal column represents the normal heart beat activity the other columns represent the different pathological states. The probability for the different concepts are announced in that way that a probability up to 50 % is green coloured, a probability over 50 % up to 80 % is orange coloured while a probability over 80 % is red coloured.

The system design regarding the time representing time axis x is variable, so the user can follow the evolutionary time history under different aspects, varying between 1 sec. for the 560 measuring points up to 6 months for the 560 measuring points.

4. Conclusions

The use of fuzzy logic and neural nets has reached a status which allow physicians creating their own patient typical classifiers in situ. Therefore, people which are not familiar with the mathematical theory of both classifiers can use this new technique empowered by soft computing methodology. This methodology will become more and more important in medical support systems for a more efficient diagnosis and supervision of patient's based on their individual and/or specific conditions.

5. References

1. Möller, D.P.F., Fuzzy Logic and Its Impact for Medical Applications, Proceedings Eufit'93, Aachen 1993, 283ff.
2. Reuter, M., A Proposed Method for Representing the Real-Time Behaviour of Technical Processes in Neural Nets, Proceedings IEEE Trans. on Systems, Man and Cybernetics, Vancouver October 1995.
3. Reuter, M., ECG-signatures analysis by adaptive FD-spectra in combination with fast conditional neural nets or Fuzzy classificators, TU Clausthal 1995.
4. Möller, D.P.F., Modelling and Simulation with Fuzzy Logic. In: Simulationstechnik (Ed.: D. Tavvangarian), Vieweg Verlag Braunschweig, 1991, 146-149.
5. Reuter, M., Adaptive FD-spectra Representations explained by motoring Diesel Motors and ECG-Signatures
6. Proceedings EUFIT'95, Aachen 1995, 192-198.
7. Reuter, M., FD-Spectra used in Medical Controlling. Proceeding EUFIT'94, Aachen 1994, 116-120.
8. Westenskow, D.R. and Loughlin, P.J., Quantitative Anaesthesia with the Help of Closed-Loop Control. In: Quantitative Anaesthesia (Eds. K. van Ackern, H. Frankenberger), Springer Verlag, Heidelberg, 1989, 109-119.

OO PHYSBE MODEL – A BENCHMARK FOR MODULAR OBJECT-ORIENTED DYNAMIC SYSTEM SIMULATION TOOLS

Matjaž Ostroveršnik[1], David Murray-Smith[2], Borut Zupančič[3], Stanko Strmčnik[4]

[1] HERMES Softlab, Litijska 51, 1000 Ljubljana , Slovenia
[2] Dept. of Electronics & Electrical Eng., University of Glasgow, Glasgow G12 8QQQ, Scotland, UK
[3] Faculty of Electrical Engineering, University of Ljubljana, Tržaška 25, 1000 Ljubljana, Slovenia
[4] Dept. of Comp. Automation & Control, Jožef Stefan Institute, Jamova 39, 1000 Ljubljana, Slovenia

Abstract. The paper proposes the Physbe (Physiological Benchmark) model (OOPhysbe) as a benchmark for today's object oriented modelling and simulation tools. Modularization and component reuse (two important indicators of object orientation) are key productivity factors in the software development process in general. The OOPhysbe benchmark was selected because it is complex enough to justify the usage of all the OO mechanisms (i.e. classes, inheritance, modularization). If a given dynamic system simulation language (DSSL) fails with the OOPhysbe benchmark model, it also fails to comply with the OO approach and therefore is not object oriented.

Introduction.

Strongly influenced by software engineering proposals, modelling and simulation tools introduced very flexible possibilities for hierarchical and modular usage many years ago [5]. Nowadays the object-oriented (OO) approach provides the highest level of component reuse and modularization. Modularization and reuse are key productivity factors in the software development process in general. Many dynamic system simulation languages (DSSLs) are said to be object oriented, but they do not fulfil all the required conditions (i.e. classes, instances, inheritance, run-time sorting, ...) in order to be regarded as true OO DSSLs. Therefore a strong need was identified for a benchmark to provide a simple test of compliance with the OO paradigm and this resulted in our Physbe (i.e. Physiological Benchmark) benchmark proposal. If a given DSSL fails to decompose the Physbe model as proposed in this paper, it also fails to fully comply with the OO paradigm.

Physbe model description

Physbe (i.e. Physiological Benchmark) [1,2] models blood flow through the human body. Each body part is considered as a blood storage compartment. There are two pumps (the right and left ventricles) that force blood around the body. The model can be decomposed into two parts. The first part, depicted in Figure 1, consists of five compartments (i.e. head, trunk, arms, legs and the so-called *InnerCycle*). All the compartments are connected in parallel. All blood goes from the InnerCycle to the external body parts (e.g., head, trunk, arms, legs) and is later returned. The *InnerCycle* is made up of four parts that are connected sequentially through valves. The *InnerCycle* structure is depicted in Figure 2. The Vena Cava collects blood from external body parts. That blood goes through the right ventricle, lungs, the left ventricle and the aorta, where it is again dispersed among external body parts.

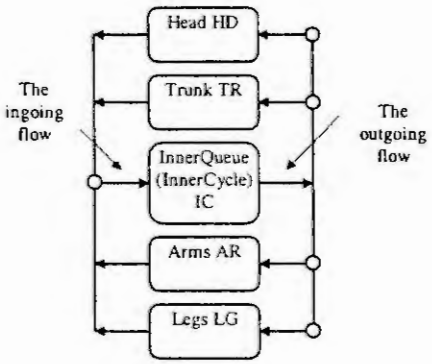


Figure 1: Body components

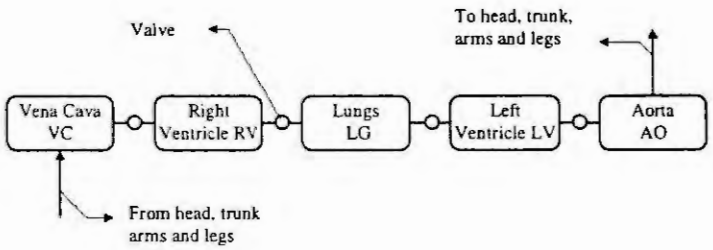


Figure 2: The inner cycle

Each of the nine body parts can be represented using a model similar to that depicted in Figure 3. The lumped compartments generally behave according to the equations obtained on the basis of mass and energy balance equilibrium:

$$\phi_i = \frac{p_i - p}{R_i} \qquad p = \frac{V}{C}$$
$$\phi_o = \frac{p - p_o}{R_o} \qquad T = \frac{H}{W}$$
$$q_d = k \cdot A \cdot (T - T_a) \qquad q_o = \phi_o \cdot T_o \qquad (1)$$
$$V = \int_0^t (\phi_i - \phi_o)dt + V_z \qquad q_i = \phi_i \cdot T_i$$
$$H = \int_0^t (q_i - q_o + q_e - q_d)dt + Hz$$

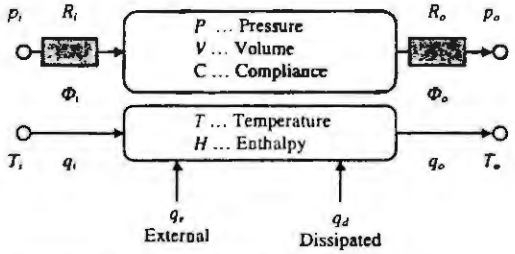


Figure 3: General lumped structure

The mnemonics used in the Physbe model are explained in Table 1.

Table 1: Explanation of the mnemonics used in the Physbe model

Symbol	Significance	Symbol	Significance	Symbol	Significance
p	pressure	Φ_i	input flow rate	VC	Vena Cava
V	volume	R_o	output resistance	AO	aorta
C	compliance	p_o	output pressure	RV	right ventricle
T	temperature	Φ_o	output flow rate	LV	left ventricle
H	enthalpy	T_i	input temperature	HD	head
p_i	input pressure	q_i	input enthalpy	TR	trunk
R_i	input resistance	q_e	external heat	AR	arms
q_d	dispersed heat	T_o	output temperature	LG	legs
q_o	output enthalpy	W	weight of the lump	LN	lungs
A	area of the lump	V_z, H_z	initial states	T_a	temperature of the ambient

Variable names in Table 1 (column 1, 2 and T_a in third column) are used in conjunction with lumped element mnemonics (third column without T_a). A general form is "<lump-name>.<variable name>" (e.g. AO.p stands for the pressure in the aorta). All symbols that are not mentioned in Table 1 are constants. Some constants are equal for all lumped elements, others have a specific value for each element. As can be seen, in each lumped element there are 9 equations of which two represent states (V and H). Unfortunately, the model defined by Eqs. 1 holds only for external body parts (head, trunk, arms and legs). Lumped elements which are part of the *InnerCycle* have slightly modified equations, due to the valves, the parallel nature of the connections with external parts and the heartbeat. The left and right ventricle are responsible for the flow of blood. The model achieves that with a variable *compliance* (pressure per unit volume) in the left and right ventricle. The signal is periodic, with a period of 1 sec. There is a linear rise from 0 to 0.4 sec, when it reaches its maximum. The rise is followed by a fast fall (0.4-0.5sec). The compliance then maintains is a small constant value until the start of the next period. The blood flow between adjacent lumped elements within the *InnerCycle* is affected by the valves. The valve is closed (the resistance is high) when the output pressure is greater than the input pressure.

Why modular Physbe?

One might ask why we selected the Physbe benchmark from all the available benchmark models. The main reasons are the following:

Size The use of the OO approach is justifiable only in cases where the number of equations is large. Small examples like the predator-prey benchmark, the aspirin dosage benchmark etc. do not offer enough complexity

and size to show the benefits of the OO approach. The Physbe benchmark has more than 100 equations. These require something more than flat ordering, and justify some syntax overhead related to model decomposition.

Clean decomposition The model can be clearly decomposed into 9 components (i.e. head, arms, legs, trunk, Vena Cava, aorta, lungs, left and right ventricles) with clean boundaries. With the 7±2 rule, which is usually used for graph drawing, the model can be further decomposed into two subgraphs. Namely the heart and lung components can be joined into the so called InnerCycle. These decompositions are depicted in Figs. 1 and 2. A side effect of clean decomposition is also a requirement for separately compiled (i.e. binary modules) and in some cases separately linked submodules. ASCII submodules are inappropriate for large models but the use of binary modules requires a run-time sorting mechanism [3].

Parallel and sequential connections Components in the top level graph (Figure 1) are connected in parallel, while submodels in the InnerCycle subgraph are connected in sequence (Figure 2). This covers all possible graph connections.

Identical submodules As mentioned in model description section two subsets (i.e. externals (head, arms, trunk, legs) and ventricles (left and right)) have identical behaviour (methods, attributes). The only differences are initialisation constants and their location in the model graph. This characteristic justifies the usage of the abstract submodel (i.e. class). Two classes (e.g. lump for external lumped elements and ventricle for both ventricles) can be formed (Figure 4). Instead of having six duplicated definitions (4+2), there are only two class definitions (i.e. lump and ventricle) and six short declarations. There is no code that is duplicated unnecessarily.

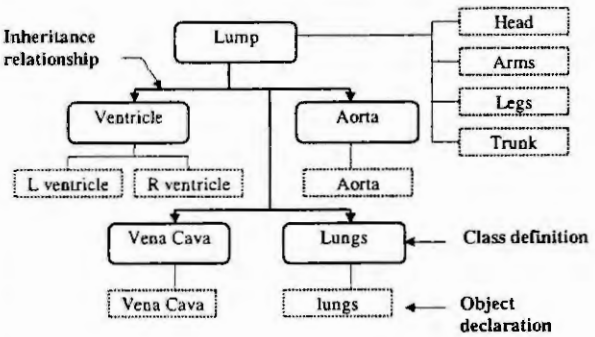


Figure 4: Hierarchical decomposition of the model

Similar submodels It was already shown that the external lumped elements and ventricles have identical behaviour. It can be further realised that other component models usually differ only in up to four equations out of 18 and in the number of external connections. Therefore it would be wise to use these 18 equations and replace or add only some of them. The ability of the language to replace the definition of an inherited method (i.e. function or procedure) is significant, since some "OO" modelling and simulation tools do not support such features (e.g. DYMOLA).

Availability of a "conventional" model description form The Physbe model is available in a number of "conventional" model description forms, e.g. ACSL model in an integral form, with minor modularization (i.e. macros). It is thus easy to compare the readability and execution of such conventional realisations with a modern OO approach.

Physbe with OOSlim

The Physbe model was implemented with our OO DSSL called OOSlim [3] (*http://www-e2.ijs.si/Matjaz.Ostroversnik/ooslim/OOSlim.htm*). Only the main features of OOSlim and the Physbe implementation will be presented.

Modularization Classes can be defined in separate files and compiled separately. The final model assembly takes place at run-time in the case of separately compiled submodules. The language also provides a run-time sorting mechanism as this is a required condition for OO DSSLs [3].

Classes definition Classes are defined in a way which is similar to those in the C++ language. One can define methods and attributes. Methods are functions and procedures that perform transformations of the class attributes. Attributes are data elements of the class. The functionality of the attributes can be modified by qualifiers. The most important qualifiers are:

calc qualifier defines an algebraic equation from the model (e.g. *calc((pi-p)/Ri)*). All attributes with *calc* qualifier together form the dynamic system (sub)model.

link and **import** qualifiers together form input data connections of the class / submodel with the environment (i.e. other submodels or simulation engine). Each class can have an arbitrary number of links with the environment, and each link can also have an arbitrary number of variable value flows.
export qualifiers define the attribute values that are available to the environment.
initial, int, and **der** attributes are used to define the initial value of state variables (e.g. *initial(Vz)*), the corresponding integration (e.g.. *int(V)*) and derivative expressions (e.g. *der(qi-qo+qe-qd)*).

Inheritance..The *Ventricle* class provides a nice example of the inheritance mechanism. It can easily be seen from the mathematical model that four out of ten attributes are calculated in a way which is different from the calculation for the *Lump* class. The calculation procedures for these attributes are redefined compared with the declaration in the *Lump* class, while other model components are left intact (i.e. used as they are, defining the superclass (e.g. *Lump* class)). In the same way three new input flows and two attributes are defined. Other classes are derived in a similar way (i.e. ventricle, lungs, Aorta and Vena Cava). The class hierarchy of the Physbe model in the OOSlim language is depicted in Figure 5. Classes in the grey area of Figure 5 are provided by the language runtime, others are application specific and therefore developed by the user. The *InnerCycle* and *Physbe* classes are discussed later on in the model topology section.

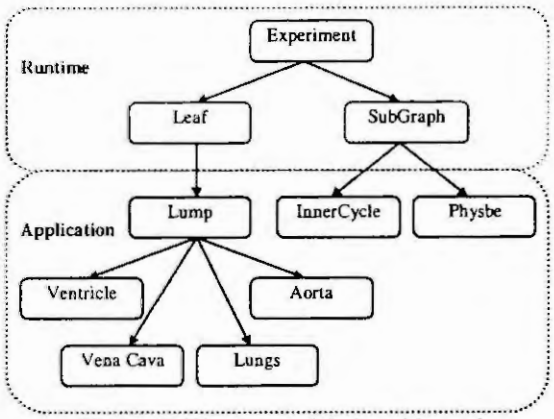


Figure 5: Physbe model class hierarchy

Model topology plays an important role if one decomposes a large model into smaller components. However all components must be joined together to behave as one unit. Model topology therefore defines the graph components and the relationships among them (i.e. the means by which they exchange information). In our example (Figure 5), there are two topology classes (i.e. *InnerCycle* and *Physbe*). Each class consists of two sections: the block and graph sections. The block section defines which components compose the submodel and it optionally declares their initial values. The graph section defines how components exchange values.

Summary

Divide and conquer and component reuse are the common rational rules used in solving complex technical problems. Dynamic system modelling techniques and associated modelling and simulation tools represent areas where such an approval can certainly be justified. The object-oriented approach offers currently the highest level of modularization and component reuse. However many recently developed tools that have been proclaimed as OO do not satisfy all the requirements for being truly object-oriented.

The OO Physbe (OOPhysbe) benchmark for evaluation of the characteristics of OO DSSLs is proposed. The OOPhysbe benchmark was selected because it is complex enough to justify the usage of all the OO mechanisms (i.e. classes, inheritance, modularization). If a given DSSL fails with the OOPhysbe benchmark model, it also fails to comply with the OO approach and therefore is not object oriented.

References

1. ACSL Reference Manual, Edition 10.1, ISBN 0-925649-04-X
2. McLeod, J., PHYSBE: A Physiological Simulation Benchmark Experiment. SIMULATION, 7 (1966), 324-329.
3. Ostroveršnik, M., Analysis of usage of object oriented paradigm for development of tools for dynamic system simulation and modelling. Ph.D. thesis, University of Ljubljana, Faculty of Computer and Information Science, 1996.
4. Ostroveršnik, M., Murray-Smith, D., Modularity in Dynamic System Simulation. Proceedings of CESA'96 IMACS Multiconference, Lille, (1996), vol 2, 660-665.
5. Zupančič, B., Modular hierarchical modelling with SIMCOS language. Mathematics and Computers in Simulation, (1998), vol. 46, 67-76.

A BLOOD FLOW MODEL BASED ON PHYSICAL CONCEPTS

Ch. Almeder[1], F. Breitenecker[1], J. Krocza[2], M. Suda[2]
[1]Vienna University of Technology
Wiedner Hauptstraße 8–10, A–1040 Vienna, Austria
[2]Austrian Research Centres Seibersdorf
A–2444 Seibersdorf, Austria

Abstract. The model of the human arterial blood flow presented in this paper is based more on physiological and physical facts rather than using heuristic assumptions or substitute models (i.e. compartment models, spring models, ...). In general this model uses a 1-1 mapping to represent the arterial network on the basis of many small flow models, each consisting of a system of first order partial differential equations. Those systems describe the hemodynamics of the human arterial blood flow in just one vessel segment in a quasi one-dimensional way. Therefore it is possible to gain detailed results in any part of the network and to extend the model by adding vessels to the network. Hence the model allows to investigate the hydraulic and hydrodynamic effects of a variety of diseases, not only valvular defects but also local and global arteriosclerosis, different kinds of arrhythmia, stenoses and occlusions of vessels, and other diseases of the cardiovascular system.

Introduction

In human medicine modelling and simulation is limited by the complexity of the investigated biological process and by the difficulties of gaining data for an individual patient [2]. The first problem — the complexity of the process — complicates the distinction between parts that must be included in the model and those that are negligible. Therefore in medical modelling a combination of black and white box models are used very often. The second problem — the limited availability of data — sometimes requires a down sizing of too complex models to smaller ones in order to simplify the parameter identification process. But also the necessity of in-vivo measurements, which are very complicated in most cases, allows only a few measurements and those are often imprecise.

Nevertheless this model is a step away from traditional heuristic substitute models towards a description based on physical and physiological concepts [3, 4]. Those physiological facts are the motivation for the identification of the arterial system with a pipe network. Every segment of an artery between two ramification points is represented by a pipe and every ramification point by a hydraulic node. In the case of no flow the arteries have a cylindrical shape and blood is lost through the walls. Furthermore it is assumed that the arteries are hydraulic smooth, and they are fixed against longitudinal stretching. It is also assumed that blood in larger vessels behaves like a Newtonian fluid.

In order to avoid the problem of determining the amount of blood flowing off at some node, a special concept of fictitious arteries is used. The model is extended by new (fictitious) arteries added to the network to collect the blood at an additional (collecting) node (see figure 1). All those arteries have the same length but variable diameters, which are used to regulate the resistance of other parts of the arterial network and the resistance of the venous system. Furthermore the microcirculation can be taken into consideration using this concept.

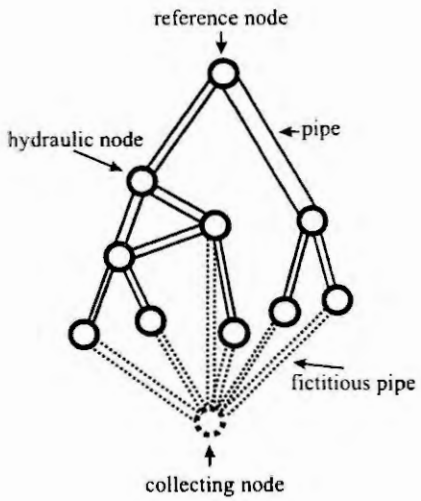


Figure 1: Representation of the arterial system as a pipe network including the fictitious arteries

Model equations

The blood flow in arteries can be described in terms of the flow velocity V and the pressure head H. In this case only the average velocity of the sectional area and the average pressure head are necessary,

because an one-dimensional model is used. Therefore V and H are functions of position x along the axis of the artery and of time t [6].

To obtain the one-dimensional continuity equation the conservation of mass is applied to a control volume, which leads to

$$H_t + \frac{a^2}{g} V_x + V H_x - V \sin\alpha = 0, \tag{1}$$

where a is the wave speed, g is the gravity, and α is the angle between the artery axis and the horizontal. It is important to notice that the wave speed is not a fixed parameter, but depends on the radius changes due to the wall elasticity

$$a = a_0 \sqrt{F\left(\frac{R}{R_0}\right)}, \tag{2}$$

$$F\left(\frac{R}{R_0}\right) = \left(\frac{R_0}{R}\right)^2 \left(1 - 2\ln\frac{R}{R_0} - \frac{g}{{a_0}^2}(H_0 - z)\right). \tag{3}$$

The equation of motion for unsteady flow can be derived by applying Newton's Second Law to a control volume of blood in the artery segment considering only the streamline direction which is parallel to the axis of the segment.

$$g H_x + V_t + V V_x + \frac{\lambda V|V|}{4R} = 0, \tag{4}$$

where R is the radius of the vessel and λ is the friction factor. The last term of (4) describes the influence of the shear stress. Again notice that λ and R depend on x and t.

The formulation of the continuity equation (1) must be extended by another relation, because it is necessary to know the actual radius R of the artery in order to evalute the function F. The elasticity model relates radius and pressure head as follows

$$\frac{R^2}{{R_0}^2}(H - z) = 2\frac{{a_0}^2}{g} \ln\frac{R}{R_0} + (H_0 - z). \tag{5}$$

Actually this equation cannot be solved for R symbolically, but by applying implicit differentiation the first terms of the Taylor sum can be calculated

$$\begin{aligned} R &= R_0 + R'(H_0)(H - H_0) + \frac{R''(H_0)}{2}(H - H_0)^2 + \ldots \\ R'(H_0) &= -\frac{1}{2}\frac{R_0}{H_0 - z - \frac{{a_0}^2}{g}} \\ R''(H_0) &= \frac{1}{4}\frac{R_0(3H_0 - 3z - 5\frac{{a_0}^2}{g})}{(H_0 - z - \frac{{a_0}^2}{g})^3} \end{aligned} \tag{6}$$

which permits a good approximation, because the relation of pressure and radius is mainly linear.

Solution method

The method of characteristics is used to develop a solution algorithm for the model equations. This method transforms the two partial differential equations into two sets of ordinary differential equations, which are called the characteristic equations [5].

$$\begin{aligned} C^+ &: \begin{cases} \frac{g}{a}\frac{dH}{dt} + \frac{dV}{dt} + \frac{V|V|\lambda}{2D} - \frac{g}{a} V \sin\alpha = 0 \\ \frac{dx}{dt} = V + a \end{cases} \\ C^- &: \begin{cases} -\frac{g}{a}\frac{dH}{dt} + \frac{dV}{dt} + \frac{V|V|\lambda}{2D} + \frac{g}{a} V \sin\alpha = 0 \\ \frac{dx}{dt} = V - a \end{cases} \end{aligned} \tag{7}$$

The equations $\frac{dx}{dt} = V \pm a$ describe the curves in the x-t-plane. Along those curves the ordinary differential equations of V and H are valid. At the intersection points of the characteristics both can be solved for V and H.

The solution of the characteristic equations can be approximated by discretisation of the problem and solving it on a x-t-grid. For that purpose the artery segment is divided into N identically of length Δx, and the points are numbered from x_0 to x_N. The second dimension of the grid – the time axis – is also divided into sections, but different intersections are possible. The time step from the t_j-line to the t_{j+1}-line is Δt_j wide.

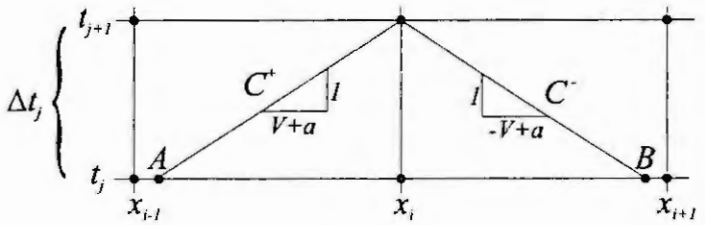


Figure 2: x-t-grid for the approximated solution

For solving the equations on the grid (see figure 2), V and H, which are given at the segmentation points, have to be interpolated at the points A and B. To calculate the velocity $V_{i,j+1}$ and the pressure head $H_{i,j+1}$ at the upper middle point, finite differences along the characteristics are used.

It is important to recognize the limitation in the selection of the grid-mesh ratio. If the time step Δt_j is increased too much, the points A and B will be outside of the range from x_{i-1} to x_{i+1}, especially at the boundaries the points would lie outside the artery. So for a fixed segmentation the time increment is limited by the *Courant* condition

$$\Delta t \leq \frac{\Delta x}{\max(a + |V|)}. \tag{8}$$

When combining several arteries, as it is done for a network, it is necessary to choose an uniform time step for the whole network, thus the equations at the connections can be evaluated simultaneously. Hence the *Courant* condition must hold for all arteries in a network.

Results

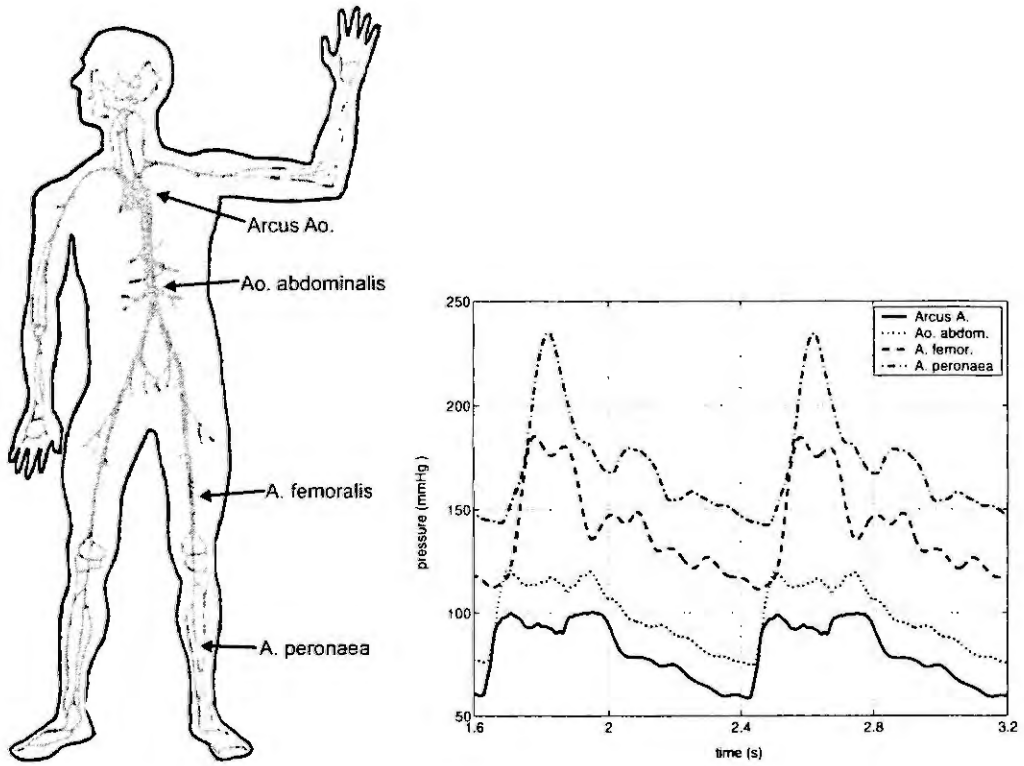
The model equations are solved on a rough overall model of the human arterial system consisting of about 150 artery segments (see figure 3(a)). The body is in an upright position as it is shown in the figure, which causes the hydrostatic pressure to affect the pulse.

There are two air chambers added to the aorta, one near the heart, the other near the bifurcation. Wave speed is set between 6 m/s at the heart and 14 m/s in peripheral arteries in the feet and hands, which means a high elasticity of the vessels — normal values for persons at young ages.

The pulse curves (see figure 3(b)) show all important qualitative and quantitative characteristics as measured arterial pressure curves described in relevant literature. Both pulse curves recorded in the aorta have a strong pressure increase at the begin of the systole, a little dicrotic notch, and a slow decrease of the pressure during the diastole. The pressure curves in the leg have a second peak due to the reflection at the closed valve and the amplitude is about double that in the aorta. Measuring the pulse in the upper arm shows a blood pressure of about 128 mmHg (sys.) – 70 mmHg (dia.), which seems a very realistic [1].

Summary

The model of the human arterial blood flow presented in this paper allows to investigate the hydraulic and hydrodynamic effects of the pulse propagation in a large vessel network based on a physical description. So an overall reaction as well as local effects of different diseases in the human arterial system can be observed.



(a) Positions for data recording

(b) Pressure curves (2 heart beats)

Figure 3: Simulation experiments on a real arterial system

In fact of the high complexity of the human arterial system the parameter identification for an individual patient is very difficult. Nevertheless it is possible to do general investigation on standardized network data in order to study the effects of different kinds of treatment and operation methods.

References

[1] Ch. Almeder. *Hydrodynamic Modelling and Simulation of the Human Arterial Blood Flow.* Phd theses, Vienna University of Technology, 1999.

[2] Ch. Almeder, F. Breitenecker, J. Krocza, M. Suda. Simulation of the humand arterial system - static and dynamic. In *Proceedings of the 1st BMES-EMBS Conference*, Atlanta, 1999.

[3] S.D. Balar, T.R. Rogge, D.F. Young. Computer simulation of blood flow in the human arm. *Journal of Biomechanics*, 22(6/7):691–697, 1989.

[4] T.J. Pedley. *The Fluid Mechanics of Large Blood Vessels.* Cambridge University Press, Cambridge, Great Britain, 1980.

[5] G.Z. Watters. *Modern Analysis and Control of Unsteady Flow in Pipelines.* Ann Arbor Science, Ann Arbor, Michigan, 1979.

[6] E.B. Wylie, V.L. Streeter. *Fluid Transients in Systems.* Prentice - Hall, Englewood Cliffs, New Jersey, 1993.

A WEB-BASED COURSE ON MODELLING AND SIMULATION – THE LAGRANGIAN APPROACH

Reinhard Gahleitner, Werner Haas, Kurt Schlacher
Department of Automatic Control
Johannes Kepler University of Linz
Altenbergerstraße 69, A-4040 Linz
reinhard[werner][kurt]@regpro.mechatronik.uni-linz.ac.at

Abstract. This paper presents some outcomes of an EU-Project called RichODL (Enriching open distance learning by knowledge sharing for collaborative computer-based modelling and simulation). One goal is to build up an internet based course on modelling and simulation of dynamical systems. Further, we present some tools, which support the user of the course within the course environment in modelling and simulation. Finally a short introduction to the theoretical background is given.

Introduction

The internet is getting more and more influence to many areas of our live. For examples in research (online-journals, online-libraries, ...), business (internet-shopping, tele-banking, ...), entertainment (chat, audio or video on demand, ...) and finally education (tele-learning, web based training, ...) to mention only a few.

In reference to the last point, we are on the way of setting up a pilot course on modelling and simulation of dynamical systems. The pilot course which, we are developing together with some partner universities across Europe, and all the knowledge-sharing infrastructure and software tools will be freely accessible via standard web browsers.

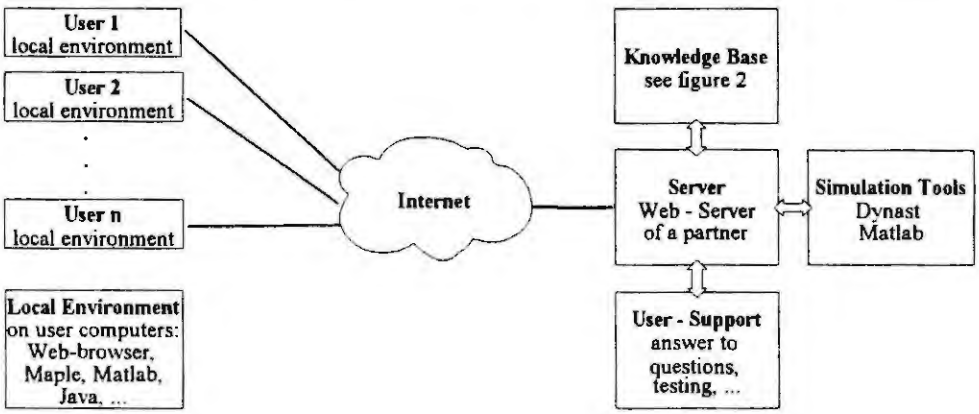


Figure 1: Usage of the online-course.

The course will be embedded into a virtual university environment to allow for assigning the course content and structure to each user individually in a way best suited to their needs, interests and preceding preparation. For this, the curriculum of the course is divided into educational units which can be arranged in a very flexible way. Beside the theoretical units which explain all the necessary basics, a very important part of the course are examples. There are many solved examples which the user can go through step by step, and a number of unsolved examples for practicing and self evaluation – see Figure 1 and 2.

The main target groups of the pilot course are regular on-campus and distance education students wishing to complement the traditional engineering courses, practising engineers seeking education and training to upgrade their qualification in their job or to start a new carrier, and teachers of engineering schools needing to upgrade the courses they teach. Also engineers in industry with some knowledge in mechanics, electromagnetics and mechatronics are targeted by this course.

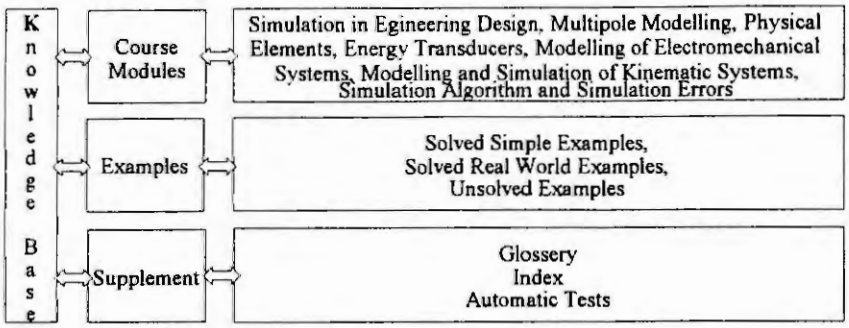


Figure 2: Overview over the course material in the knowledge base.

Course Environment

One very important part of the course is the possibility for a user to set up and solve own examples. For that some software packages (Java enabled browser, Matlab [4], Maple [5]) are needed on the local computer. Further, the user is supported with software tools for the derivation of the mathematical models in a symbolic way, generation of simulation code (link to Dynast [3] and Matlab/Simulink) and simulation. These tools can be loaded form our RichODL page [2]. Some other utilities (knowledge base, examples, remote simulation tools, ...) are available for online use – see figure 1 and also [3], [7].

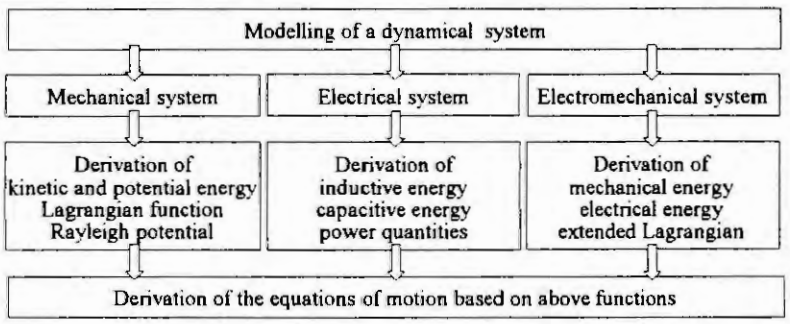


Figure 3: Modelling process for electromechanical systems.

We have developed tools in Maple V, that give aid in generating the desired energy and potential functions, and there are functions for deriving the equations of motion in a fully automatic way – see Figure 3. The implicit second order differential equations from the previous step are then converted to

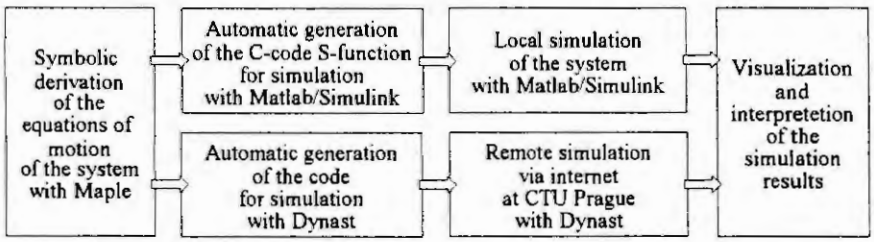


Figure 4: How to run the modelling and simulation.

a system of explicit first order differential equations of the form $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$ if this is possible. Another set of functions generate ready-to-use C-Code (the so-called S-function) for simulation in Simulink or simulation-code for Dynast. Finally the simulation can be done in two ways – see Figure 4. First, for local simulation on the desktop computer Simulink can be used and from this all its standard tools and toolboxes are available. The second possibility is remote simulation with Dynast. A simulation request

is sent with a special HTML-page from a standard web browser to the Dynast server in Prague [1], and the results come back and are displayed within the browser. The visualization of the results can be done in form of diagrams in standard HTML-pages or within Java-applets.

Modelling of Electromechanical Systems

The topic of our modules is modelling of electromechanical systems with lumped parameters based on an energy approach. The first step in analyzing an electromechanical system by a conservation of energy approach is to reduce the system containing electromechanical coupling terms to a minimum. To do this, separate out all purely electrical parts and all purely mechanical parts of the system including losses. We suppose that the electrical network consists of n inductances L, k capacitors C, and static elements S (resistors, sources). Moreover, the currents $i_i, i \in \underline{L}$ through the inductances and the voltages $u_i, i \in \underline{C}$ along the capacitors are the electrical coordinates. Networks which satisfy this property are called simple – [6]. The separation procedure results in the general conservative electromechanical coupling network

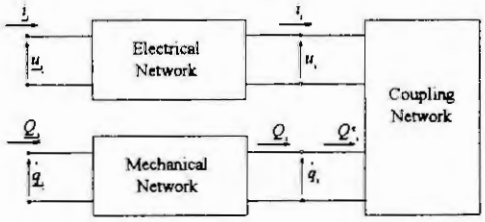


Figure 5: Simplification of electromechanical systems.

in Figure 5 with electrical and mechanical terminal pairs. So, it is possible to consider each mechanical terminal pair individually to find the generalized force due to the electromechanical coupling. Let us define the generalized force Q_i^e as the force applied to the ith mechanical coordinate by the coupling network. Q_i^e can be found by considering that an arbitrary placement $\mathrm{d}q_i$ of the ith generalized mechanical coordinate during the time $\mathrm{d}t$ takes place. All other mechanical coordinates are fixed and the electrical coordinates may change in accordance to the internal constraints due to the electrical network. During the displacement the conservation of energy must hold. which leads to the generalized electromechanical coupling force Q_i^e applied to the ith terminal either by the magnetic field or by the electrical field

$$\begin{aligned} Q_i^e &= \frac{\partial W_m^{\mathrm{i}}}{\partial q_i}, \qquad \text{with} \qquad W_m^{\mathrm{i}} = \int_{0,\dots,0}^{i_1,\dots,i_n} \sum_{i=1}^{n} \psi_i \left(i_1', \dots, i_n'; q_1, \dots, q_m\right) \mathrm{d}i_i' \\ Q_i^e &= \frac{\partial W_e^{\mathrm{u}}}{\partial q_i}, \qquad \text{with} \qquad W_e^{\mathrm{u}} = \int_{0,\dots,0}^{u_1,\dots,u_l} \sum_{i=1}^{l} \bar{q}_i \left(u_1', \dots, u_l'; q_1, \dots, q_m\right) \mathrm{d}u_i'. \end{aligned} \tag{1}$$

W_m^{i} and W_e^{u} are called the total magnetic coenergy of all inductances and the total electrical coenergy of all capacitances, respectively. The associated flux linkages and charges are denoted by ψ and $\bar{q}$, respectively. The prime indicates the variable of integration. Finally, the equations of motion of the mechanical part have the form

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L^{\mathrm{ex}}}{\partial \dot{q}_i} - \frac{\partial L^{\mathrm{ex}}}{\partial q_i} + \frac{\partial P^{\mathrm{R}}}{\partial \dot{q}_i} = Q_i, \qquad \forall i, \qquad L^{\mathrm{ex}} = T + W_e^{\mathrm{u}} + W_m^{\mathrm{i}} - V, \tag{2}$$

with the extended Lagrangian L^{ex}, the total dissipated mechanical power P^{R} related to a Rayleigh function, the generalized external forces Q_i, and the set of generalized mechanical coordinates $(q_1, \dots, q_m)$ – see [6]. Moreover, the equations of motion of the electrical part have the form

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial}{\partial u_i} W_e^{\mathrm{el}} + \frac{\partial}{\partial u_i}\left(P^{\mathrm{L}} + P^{\mathrm{u}}\right) = 0, \quad i \in \underline{C}, \qquad \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial}{\partial i_i} W_e^{\mathrm{el}} + \frac{\partial}{\partial i_i}\left(P^{\mathrm{C}} + P^{\mathrm{i}}\right) = 0, \quad i \in \underline{L} \tag{3}$$

with $W_e^{\mathrm{el}} = W_e^{\mathrm{u}} + W_m^{\mathrm{i}}$. The power quantities $P^{\mathrm{C}}, P^{\mathrm{L}}, P^{\mathrm{u}}$ and P^{i} are given by

$$\begin{aligned} P^{\mathrm{C}}(u_1,\dots,u_l;i_1,\dots,i_n) &= \sum_C ui, & P^{\mathrm{L}}(u_1,\dots,u_l;i_1,\dots,i_n) &= \sum_L ui, \\ P^{\mathrm{u}}(u_1,\dots,u_l;i_1,\dots,i_n) &= \sum_S P^{\mathrm{u}}(u), & P^{\mathrm{i}}(u_1,\dots,u_l;i_1,\dots,i_n) &= \sum_S P^{\mathrm{i}}(i) \end{aligned} \tag{4}$$

and the are expressed in terms of the generalized electrical coordinates. The power quantities $P^{\mathrm{u}}(u)$ and $P^{\mathrm{i}}(i)$ are defined for static elements S such that the relations

$$\frac{\partial}{\partial u} P^{\mathrm{u}}(u) = i, \qquad \frac{\partial}{\partial i} P^{\mathrm{i}}(i) = u \qquad \text{and} \qquad P^{\mathrm{u}}(u) + P^{\mathrm{i}}(i) = ui \tag{5}$$

are met. In the equations (4) arise the restrictions, which are given from Kirchhoff's laws. The following simple example illustrates the approach. The system consists of a capacitor realized by two plates, one

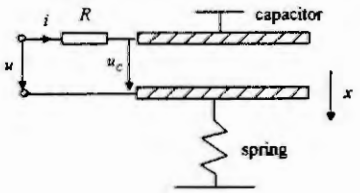


Figure 6: Example of an electromechanical system.

is fixed and the other is movable but attached to a spring – see Figure 6. The generalized coordinates are the position x at the equilibrium of the plate and the voltage u_C along the capacitor. Further, the capacitance is given as $C(x) = k/x$. The energy and power terms can be calculated as

$$W_e^{\mathrm{u}} = \frac{ku_C^2}{2x}, \quad W_m^{\mathrm{i}} = 0, \quad T = \frac{1}{2}mv^2, \quad V = \frac{1}{2}cx^2, \quad P^{\mathrm{R}} = 0, \quad P^{\mathrm{L}} = 0, \quad P^{\mathrm{u}} = \frac{(u-u_C)^2 R}{2} \tag{6}$$

which determines the extended Lagrangian L^{ex} and finally the equations of motion

$$m\frac{\mathrm{d}v}{\mathrm{d}t} + \frac{ku_C^2}{2x^2} + cx = 0, \qquad \frac{k}{x}\frac{\mathrm{d}}{\mathrm{d}t}u_C - \frac{ku_C}{x^2}v - (u-u_C)R = 0, \qquad \frac{\mathrm{d}x}{\mathrm{d}t} = v. \tag{7}$$

Summary and Outlook

We have presented a survey of an online course in the internet and also some tools which we have developed for that purpose. In a next step we will embed the material into a virtual university environment and give a pilot course to a group of students. On the other hand we are working to improve the software tools and to enable remote simulation also with Matlab/Simulink via a suitable Java-interface.

References

1. RichODL page at CTU Prague: http://icosym-nt.cvut.cz/odl/
2. RichODL page at JKU Linz: http://regpro.mechatronik.uni-linz.ac.at/eu-projekt/RichODL_Linz.htm
3. Dynast Server at CTU Prague: http://icosym-nt.cvut.cz/dyn/
4. Matlab and Simulink homepage, Mathworks Inc.: http://www.mathworks.com/
5. Maple homepage, Waterloo Maple Inc.: http://www.maplesoft.com/
6. Schlacher, K., Kugi, A., Scheidl, R., Tensor Analysis Based Symbolic Computation for Mechatronic Systems, Mathematics and Computers in Simulation, Vol. 46, 517-525, 1998
7. Mann, H., A Web Based Course on Modelling and Simulation - The Multipole Approach. 3rd MATHMOD, Vienna, 2000, to appear

A WEB-BASED COURSE ON MODELING AND SIMULATION – THE MULTIPOLE APPROACH

H. Mann[1]
Czech Technical University in Prague
CZ-166 35 Prague, mann@vc.cvut.cz

Abstract. This contribution outlines the part of a web-based pilot course on modeling and simulation of multidisciplinary engineering systems focused on the multipole approach to system modeling. Once formed, multipole models of real system modules can be reused to form complete models of complex systems in a kit-like way based on mere inspection of the systems without the need for any equation formulation or graph construction. The multipoles can be characterized by a multipole model of the internal structure of the related module, by equations or by measured data. The course is supported by a remote simulation engine and a knowledge-sharing toolset. The simulation engine not only solves the equations representing the system models, but also formulates them automatically.

Introduction

Only few engineering schools introduced modeling and simulation as a distinct topic and gave their students an opportunity to deal with real-life problems. Professors often assume that simulation is just a matter of routine utilizing a ready-made software package and that this activity requires only reading the package manual. As they consider this uninteresting academically, many of them still have had no personal experience in simulation that they could share with their students.

In some universities, courses on modeling and simulation are given to engineering students by mathematicians. Such courses usually address only part of the problem, as the mathematicians are mostly interested only in the equations resulting from modeling without questioning their validity. In addition, mathematically oriented faculty members, elated over the beauty of infinitesimal calculus, often over-emphasize analytical methods for solving the equations in a closed-form by means of symbolic manipulations. Using this approach, only small linear and a handful of special nonlinear differential equations can be solved. That leaves out the vast majority of engineering problems.

The equations practicing engineers are faced with, are not well defined, their accuracy is uncertain, and engineers do not have enough time (and qualification) to investigate them thoroughly. An engineer needs to investigate various modifications of his dynamic-system model by formulating and solving many sets of equations in a very limited time. Instead of the 'best' numerical method for each particular set of equations, he prefers a method that is sufficiently robust and applicable to a large variety of equations.

One might expect that modeling and simulation is a topic covered fully in control-engineering courses. A closer look reveals, however, that the courses are usually concerned only about 'functional models' suitable for control synthesis. Such models are very approximate as their complexity is constrained by the capability of the available control-synthesis methods. Expressions describing the functional models are usually formulated 'by hand' and evaluated using block-diagram oriented software tools like Simulink, for example.

But control synthesis makes just one step in the engineering design procedures applied to real-life systems in the industrial environment. Within these procedures, the control-synthesis outcomes must be verified and revised in an iterative way using system prototypes. The industry tends to replace the construction of real prototypes and their experimental verification by less expensive and time-consuming 'virtual prototyping' based on simulation experiments. The virtual prototypes in the form of 'physical models' must mimic dynamic behavior of the designed systems much more realistically than the functional models suited to control design and any block-diagram oriented tools are insufficient for this task.

Many engineers in large industrial organizations are very competent in the field of virtual prototyping. Unfortunately, they very rarely publish their knowledge. Newcomers in large organizations can fill the gaps in their education and training by learning from old-timers, but those working in small enterprises have to struggle on their own. To help them, a web-based course on modeling and simulation of multidisciplinary engineering systems is developed within the two-year EU Socrates ODL project [1].

---

[1] This research was done as a part of the Socrates ODL Transnational Cooperation Project No. 56057-CP-1-98-1-CZ-ODL-ODL.

Multipole modeling

Virtual prototyping requires appropriate computational methods and software tools capable not only of solving equation characterizing the physical models, but also of formulating them based on the physical laws governing energy interactions in the modeled systems. Besides this, virtual prototyping requires a unified multidisciplinary approach to physical modeling as the modern machines, instruments and other engineering systems employ simultaneously mechanical, electrical, fluid, control, and other components coming from diverse engineering disciplines. Using block diagrams for physical modeling is too laborious, time consuming and error prone. Before a computer can be used to analyze the block diagram the equations must be formed 'by hand' first, and the corresponding block diagram must be constructed, 'by hand' again. In addition, the block-diagram approach is impeded by the 'causality', 'algebraic loop', 'change-of-the-order' and other problems.

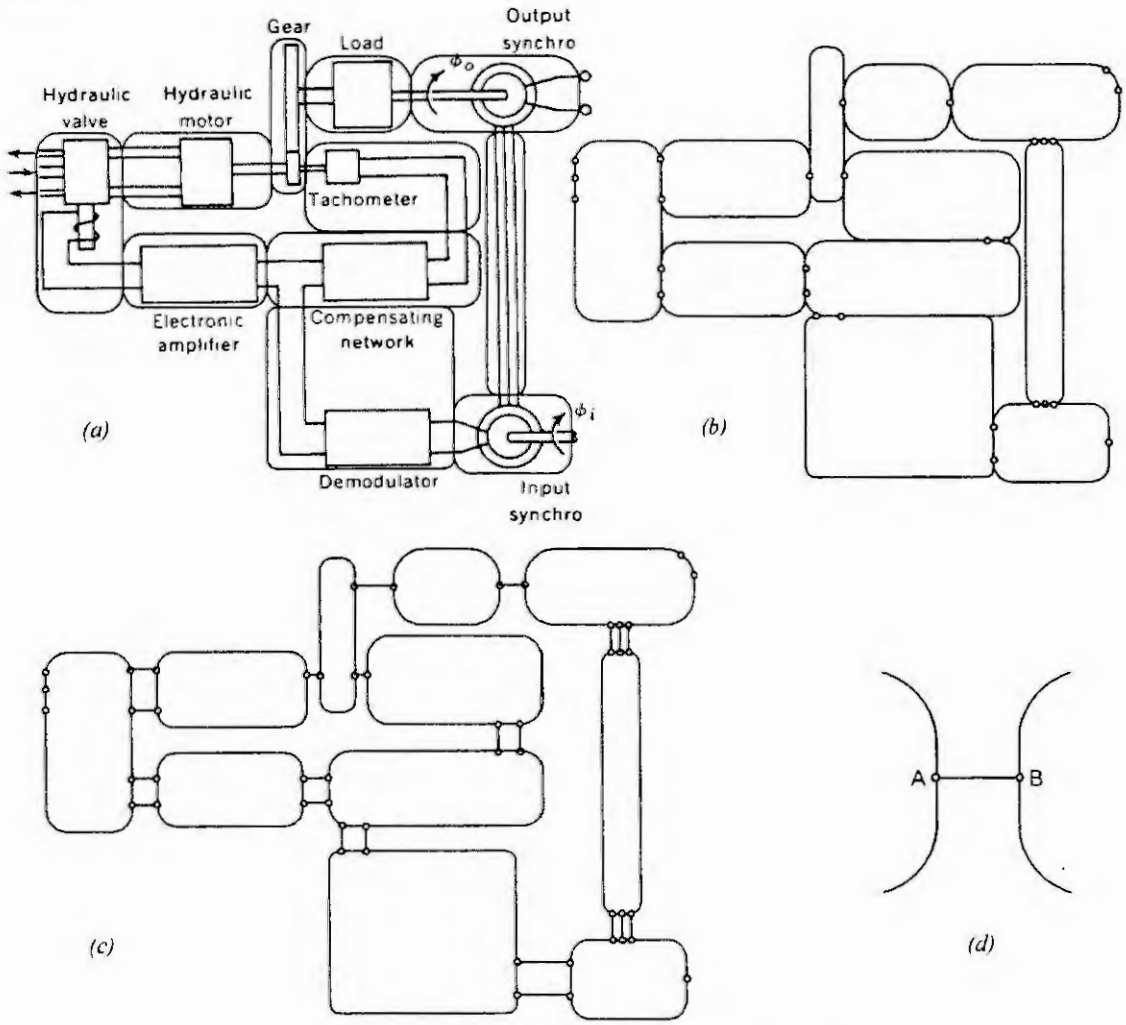


Figure 1: Multipole model of a multidisciplinary system.

Modeling of a dynamic system is based on the system reticulation, or space discretization. The system under study is considered as decomposed into a number of real subsystems or *system modules*. The dynamic behavior of a real system is governed by the flow, storage, and interchange of various forms of energy among the system modules and between the modules and the system surroundings. In order to specify the module interactions in a rigorous way, we should consider each module separated from the other system modules and from the system surroundings in the system by a closed real or imaginary surface. This *module boundary* should be chosen in such a way that it completely encircles the module without cutting into the boundary of any other module. Fig. 1*a* shows an example of a multidisciplinary system decomposed into modules separated by closed curves denoting the module boundaries.

In general, to compute the total energy interaction of a module, we should integrate all the infinitesimal energy flows all over its boundary surface. In many practical cases, however, we can assume that the energy interactions are constrained only to a limited number of regions in the boundary surface called *energy entries*. These might be cross-sections of interconnecting electrical conductors, of pipes or ducts carrying a fluid, of mechanical links or shafts, of heat-transferring contacts, etc. Fig. 1*b* shows the boundaries of the modules shown in Fig. 1*a*. The module energy entries are denoted there by small circles.

A further step taken in many practical applications is that the physical quantities across the region of each energy entry is approximated to be uniform. That is, the quantities are assumed to have bulk average values over the energy entry in the boundary. As a result, the energy flow through each entry is approximated by the product of two bulk physical quantities – *power variables* – associated with the entry. They complement each other in the sense that one of them in each pair is always a *through variable*, while the other one is an *across variable*. The criterion for determining whether a certain physical quantity is either a through or an across variable can be based unambiguously on the way in which it can be directly measured.

To explain this point more clearly, Fig. 1*c* shows the module boundaries from Fig. 1*b* detached. The interactions between module energy entries are denoted there by *ideal connections*. To measure through variables between entries (like electrical currents, forces, volume flows, entropy flow rates, etc.), their ideal connection must be replaced by an appropriate measuring instrument. The across variables of the entries (like electrical voltages, velocities, pressures, temperatures, etc.) should be measured by instruments connected between the entries and the related references (like the absolute frame, electrical ground, open-space pressure, zero temperature, etc.).

Multipoles are module models resulting from the space discretization based on the approximations outlined above. Fig. 1*c* can be considered as an example of a multipole diagram of the dynamic system shown in Fig. 1*a*. The small circles denote there multipole *poles*. Let us assume that the j-th pole of an n-pole N is associated with the pole-across variable v_j and with the pole-through variable i_j oriented towards N, $j = 1, 2, \ldots, n$. Then these $2n$ pole variables of N are interrelated by the following four postulates.

I. *Postulate of power intake*

$$\sum_{j=1}^{n} v_j i_j = 0$$

II. *Postulate of continuity*

$$\sum_{j=1}^{n} i_j = 0$$

III. *Postulate of compatibility*

$$v_{jk} = v_j - v_k \qquad j,k = 1, 2, \ldots, n$$

where v_{jk} is the across variable of the j-pole with respect to the k-th pole of N

IV. *Constituvity postulate*

$$\boldsymbol{f(v, i, t) = 0}$$

where $\boldsymbol{f(\,.\,)}$ is an n-dimensional functional, $\boldsymbol{v}$ and $\boldsymbol{i}$ is an n-dimensional vector of pole-across and pole-through variables of N, respectively

If the poles A and B are interconnected by an ideal connection as shown in Fig. 1*d*, the pole variables of the poles are constrained by the following two relations

$$v_A = v_B \qquad\qquad i_A + i_B = 0$$

These relations introduced by the interconnections as well as the postulates I through III are independent of the constitutive relation of N. As a result, the multipole approach is of the following advantages:

- system models can be formed in a kit-like way based on mere inspection of the real systems model(s) for each module can be developed debugged, tuned up and validated once for ever and then reused any time if needed
- this job can be done for each module by an expert in the related field
- each model can be represented using a different approach suiting best to the related engineering discipline or applications, i.e. by a multipole model of the module internal structure, by equations of various forms, or by measured data

- for a given module, models of different modeling abstraction and idealization can be easily replaced one by another
- the model of a module can be easily replaced by a model of another similar module (e.g., model of a rotational hydraulic motor by a model of an electrical motor)

Remote simulation

The course learners can solve their problems using the DYNAST package implemented on a server at CTU Prague and accessible via the Internet [2]. The dynamic systems under investigation can be characterised by a multipole physical model, by a set of nonlinear algebro-differential equations in a natural textual form, by a block diagram (with any number of 'algebraic loops'), or by a combination of these three approaches. Block and multipole diagrams can be submitted using a schematic capture editor implemented in the form of a Java applet. For an even more comfortable approach, DYNAST user's environment can be downloaded from the server [2].

Knowledge sharing

As mentioned already, in large industrial or research organisations engineers can learn from each other in an informal way – or in other words – by *organisational learning*. This can be defined as a process by which knowledge, that is created or made explicit during work on tasks, is captured, structured, maintained, and evolved so it can be accessed and delivered when needed to inform future tasks. In the field of modeling and simulation of multidisciplinary engineering systems, the project should give a similar opportunity to those outside large organisations. A conceptual framework for organizational learning based on communication via hypermedia enriched by knowledge sharing enhances the course to reuse previous problem solutions and capture new solutions.

The knowledge sharing toolset is developed by the Knowledge Media Institute of the Open University, U.K. The kernel of the toolset is formed by a 'case memory' which is a server supporting capturing, structuring and maintaining knowledge created during work on tasks. The students are given an access to the case memory content to support their learning. The underlying knowledge models serve to determine which items in the memory are relevant to their problem. A reasoning mechanism interprets and tailors the selected knowledge to relate it to the student's current task. This knowledge is actively delivered and presented to the student in a way enhancing the student's learning process. Errors in the solution submitted by the student are recognised by a critiquing tool. Consequently, the students can learn from (a) the store of previously solved problems, (b) from the other students in a current group, (c) from the corrections to the errors they made. The solution produced by a student based upon this 'learning by working' mechanism is in turn fed back into the case memory, and after a memory reorganisation, it is stored there for future re-use.

Students are able to communicate with the server via WWW using 'enriched presentations', i.e., by hypermedia underlined by knowledge. Any common WWW browser can be used for this purpose as the client tool is implemented in the form of a Java script. This allows for embedding the communication directly into the WWW pages of the 'virtual university' learning environment, which is used also for the delivery of the conventional part of the course. The course material is divided into learning units so that – thanks to the virtual university option – the course content and structure could be assigned individually to each student in a flexible way best suited to their needs, interests, and prior experience.

References

[1] *RichODL – Enriching ODL by knowledge sharing for collaborative computer-based modelling and simulation.* Socrates ODL Transnational Cooperation Project No. 56057-CP-1-98-1-CZ-ODL-ODL http://icosym-nt.cvut.cz/odl/

[2] Website of the *Virtual Action Group on Multidisciplinary System Simulation*, a part of the IEEE Control Systems Society Technical Committee on Computer Aided Control System Design http://icosym.cvut.cz/cacsd/msa/onlinetools.shtml

[3] R. Gahleitner, W. Haas, K. Schlacher: *A Web-Based Course On Modeling And Simulation -- the Lagrangian Approach.* In these Proceedings of the 3rd MATHMOD Symposium.

OPTIMIZATION OF FEED RATE PROFILES IN FED-BATCH BIOREACTORS WITH RESPECT TO PARAMETER ESTIMATION: HEURISTIC VERSUS PURELY NUMERICAL CONTROL PARAMETERIZATION

K. J. Versyck[1], J. R. Banga[2], E. Dens[3], and J. F. Van Impe[4,*]
[1,3,4] BioTeC – Bioprocess Technology and Control
Department of Food and Microbial Technology, Katholieke Universiteit Leuven
Kardinaal Mercierlaan 92, B-3001 Heverlee (Belgium) Tel.: +32-16-32.19.47 Fax.: +32-16-32.19.60
E-mail: jan.vanimpe@agr.kuleuven.ac.be (* Corresponding author)
[2] Chemical and Food Engineering Lab, Instituto de Investigaciónes Mariñas (C.S.I.C.)
C/Eduardo Cabello 6, 36208 Vigo (Spain)

Abstract. Unstructured models prove to be an efficient tool for (model-based) bioreactor optimization, monitoring, and control. In this contribution we present several approaches for the input design problem for experiment(s) aimed at parameter estimation of an unstructured microbial growth model. The volumetric substrate feed rate into a fed-batch bioreactor is selected as the control input to be optimized with respect to a criterion which quantifies the best attainable parameter estimation quality. It is shown that this *optimal control problem* must be solved by (*i*) *parameterization of the feed rate-time profile*, and subsequently, (*ii*) *parametric optimization* of the resulting functional structure (of the feed rate-time profile). As such, the optimal control problem is reduced to a *finite dimensional nonlinear optimization* problem. A comparison is made between two approaches. In the first (heuristic) approach, the exploitation of prior knowledge on the optimal feed rate profile for process performance leads to optimal control solutions by the optimization of only two degrees of freedom. In the second (purely numerical) approach, optimal control solutions are obtained by the parametric optimization of steps or ramps feed rate profiles. Deterministic, stochastic, and hybrid (deterministic-stochastic) programming codes have been used to solve the latter parametric optimization problem.

Introduction

This paper deals with experiment design aimed at parameter estimation for an unstructured model describing the growth of a single microbial species on one limiting substrate. It is assumed that the model structure is correct. Hence, the system identification problem is reduced to the problem of estimation of the parameters from experimental data. Often experimental data sets do not contain the pertinent information needed for high quality estimation of the kinetic coefficients which typically leads to low accuracies and high correlations associated with the estimates. Careful experiment design is needed to improve the parameter estimation quality. In this paper, the optimal experiment design problem is formulated as an optimal control problem. Two control parameterization strategies have been established for solving this optimal control problem (which cannot be solved by an indirect solution procedure). Apart from the heuristic approach which has been introduced elsewhere [5], a purely numerical approach is considered in which piecewise constant or piecewise linear control profiles are considered for parametric optimization.

Mathematical model for fed-batch bioreactor

Bioprocesses in which one biomass is growing on one limiting substrate in a perfectly mixed fed-batch bioreactor can be described by the following set of differential equations:

$$\begin{aligned}
\frac{dC_S}{dt} &= -\sigma(C_S)\,C_X + \frac{F_{in}}{V}\,(C_{S,in} - C_S) \\
\frac{dC_X}{dt} &= \mu(C_S)\,C_X - \frac{F_{in}}{V}\,C_X \qquad \text{with } C_X(0) = \frac{X(0)}{V(0)} \\
\frac{dV}{dt} &= F_{in} \qquad \text{with } V(0) = \frac{V_* C_{S,in}}{C_{S,in} - C_S(0)}
\end{aligned} \tag{1}$$

with C_S the concentration of substrate (S denotes the absolute substrate amount), C_X the biomass concentration (X denotes the biomass amount), V the volume of the liquid phase, $C_{S,in}$ the substrate concentration in the influent, F_{in} [L/h] the volumetric feed rate, V_* the initial volume without substrate, σ [g/g DW h] the (overall) specific substrate consumption rate and μ [1/h] the (overall) specific growth

rate. Note that the initial substrate amount is obtained by supplying a substrate solution with the concentration $C_{S,in}$. The rates σ and μ are defined as follows:

$$\sigma(C_S) = \frac{1}{Y_{X/S}}\mu(C_S) + m \quad \text{and} \quad \mu(C_S) = \mu_m \frac{C_S}{K_p + C_S + C_S^2/K_i} \tag{2}$$

with $Y_{X/S}$ the biomass on substrate yield coefficient, m the (overall) specific maintenance demand, K_p the parameter indicating how fast the optimum for the specific growth rate μ is reached, and K_i the inhibition parameter. The non-monotonic expression for the specific growth rate μ –see right-hand expression in (2)– is known as the Haldane growth law. The parameters, operation conditions, and initial conditions used during simulations are listed below (where u.d. denotes user-defined).

μ_m	2.1	[1/h]	m	0.29	[g/g DW h]	$C_{S,in}$	500	[g/L]	$X(0)$	10.5	[g DW/h]
K_p	10	[g/L]	$Y_{X/S}$	0.47	[g DW/g]	U_{MAX}	1	[L/h]	$S(0)$	u.d.	[g]
K_i	0.1	[g/L]				V_{MAX}	10	[L]	V_*	7	[L]

For calculation of the weighting matrix $\mathbf{Q}$ in the information matrix (as defined in (3)) following covariances of the measurement errors of the substrate concentration C_S and the biomass concentration C_X are used: $\sigma^2_{C_S} = 1 \cdot 10^{-2}$ g^2/L^2 and $\sigma^2_{C_X} = 6.25 \cdot 10^{-4}$ g^2/L^2.

Optimal control problem statement

When considering the substrate feed rate into a fed-batch bioreactor as the control input u, the experiment design problem for parameter estimation can be formulated as the following optimal control problem: *find an admissible history of the feed rate* $u(\equiv F_{in})(t)$ *minimizing a scalar function of the Fisher information matrix*:

$$\min_{u(t)\in\mathcal{U}} scal(\mathcal{F}) \text{ with } \mathcal{F} \triangleq \int_0^{t_f} \left(\frac{\partial \mathbf{y}}{\partial \mathbf{p}}(t)\Big|_{\mathbf{p}_o}\right)^{\mathrm{T}} \mathbf{Q} \left(\frac{\partial \mathbf{y}}{\partial \mathbf{p}}(t)\Big|_{\mathbf{p}_o}\right) dt \tag{3}$$

In this case, the output vector $\mathbf{y}^T$ is defined as $[C_S\ C_X]$ and the vector of parameters to be estimated $\mathbf{p}^T$ as $[K_p\ K_i]$ (see (2)). The third kinetic coefficient μ_m is assumed to be known. The weighting matrix $\mathbf{Q}$ is the inverse of the covariance matrix for the zero-mean gaussian white measurement noise. The matrix $\partial\mathbf{y}/\partial\mathbf{p}$ contains the model output sensitivities evaluated along the nominal output trajectories $\mathbf{y}_o$. Apart from the dynamic constraints imposed by the state equations in (1), the admissible regions $\mathcal{X}$ and $\mathcal{U}$ for the states and the input respectively are defined as:

$$\mathcal{X} : \forall t \in [0, t_f], i = 1,2,3 : x_i(t) \geq 0 \quad \wedge \quad \mathcal{U} : \forall t \in [0, t_f] : 0 \leq u(t) \leq U_{MAX} \tag{4}$$

As a case study the minimization of the condition number of the information matrix $\mathcal{F}$ –known as modified E-optimal design, see e.g., [3]– is aimed at:

$$scal(\mathcal{F}) \triangleq \Lambda(\mathcal{F}) = \frac{\lambda_{max}(\mathcal{F})}{\lambda_{min}(\mathcal{F})} \text{ with } \lambda \text{ an eigenvalue of the matrix } \mathcal{F} \tag{5}$$

The motivation for chosing this criterion in this study is twofold.
1. *The minimum of the modified E-cost is known exactly.* By definition the minimum of the condition number of the matrix $\mathcal{F}$ equals unity. The *prior* knowledge of the minimum of the cost (5) is in particular interesting when evaluating the optimization results obtained. When the cost equals unity the solution corresponds to a global minimum, i.e., it is an optimal control solution.
2. *The criterion can geometrically be interpreted.* A value of unity for the cost (5) corresponds to circular lines of constant identification functional $\mathcal{J}_I$ (6) values in the parameter plane:

$$\mathcal{J}_I(\mathbf{p}) \triangleq \int_0^{t_f} (\mathbf{y}(\mathbf{p},t) - \mathbf{y}_m(t))^{\mathrm{T}} \mathbf{Q} (\mathbf{y}(\mathbf{p},t) - \mathbf{y}_m(t)) dt \tag{6}$$

in which $\mathbf{y}_m$ is the vector of measured outputs and $\mathbf{y}(\mathbf{p})$ is the vector of model predictions by using the parameter vector $\mathbf{p}$. Since the optimal experiment design methodology basically relies on a linearization of the identification functional $\mathcal{J}_I$ around a nominal parameter set $\mathbf{p}_o$, the minimum of the modified E-cost (5) implies that in the neighborhood of the nominal set $\mathbf{p}_o$ a complete decorrelation between the parameter estimates is achieved (for more details reference is made to, e.g., [3]).

Solution procedures for the optimal control problem

In recent literature on dynamic optimization, two classes of methods for solving optimal control problems are distinguished (apart from the dynamic programming methodologies).

Indirect approach of the optimal control problem. Indirect approaches exploit the necessary conditions of Pontryagin (*minimum principle*) to yield a Two Point Boundary Value Problem (TPBVP) [2]. This principle starts from the fact that in general a cost function can be written as the sum of (*i*) a scalar algebraic function of the final state vector and the final time (*terminal cost*), and (*ii*) an integral of a scalar function of the state vector, the control vector, and time (*integral cost*). The scalar cost functions of the matrix $\mathcal{F}$ in the optimal control problem under study (3) are *linear* or *nonlinear* combinations of the matrix elements in $\mathcal{F}$. Furthermore, these matrix elements are integrals over time. By consequence, the objective function $scal(\mathcal{F})$ can only be written as an integral cost when it is a *linear* combination of these matrix elements. Observe this is not the case for the modified E-cost defined by (5). Nevertheless, the scalar functions in (3) can all be formulated as terminal costs. However, they are *non-differentiable* since the integrands of the matrix elements in $\mathcal{F}$ are nonlinear combinations of the sensitivity functions for which no analytical expressions are available. Consequently, the optimal experiment design problem cannot be converted into a TPBVP, i.e., this problem cannot be solved –neither analytically, nor numerically– by this indirect optimal control solution approach.

Direct approach of the optimal control problem. In the direct approaches the original problem is transformed into a nonlinear programming problem, either using control parameterization, or complete (control and state) parameterization. In this section we consider two solution procedures via control parameterization. The infinite dimensional dynamic optimization problem is reduced to a finite dimensional problem by the following consecutive steps: (*i*) definition of the structure of the input-time profile as an assembly of –preferably, simple basis– functions, and (*ii*) parametric optimization of this structure.

Heuristic approach. This approach is extensively elaborated in [5], and relies on the following Conjecture: *A feed rate strategy which is optimal in the sense of process performance, is an excellent starting point for feed rate optimization with respect to estimation of those parameters with large influence upon process performance.* The control input is constructed based on *prior knowledge* and *theoretical analysis* of the optimal feed rate profile for process performance. For several feed rate strategies constructed as such the parametric optimization of the corresponding two degrees of freedom results in complete decorrelation of the parameter estimates as quantified by the (a priori known) minimum value of 1 for the modified E-design cost (5), i.e., these feed rate profiles are optimal control solutions (in the sense of Pontryagin's minimum principle). This approach can be classified as a direct approach. However, by the heuristically motivated choice of particular basis functions, the search space is considerably reduced.

Numerical approach with extensive control parameterization. The original optimal control problem (5) is transformed into a nonlinear programming problem via approximation of the control profiles by a finite family of relatively simple basis functions. By refining the parameterization the optimal control solution can be approximated better and better. As such, the optimal control problem can be solved in a pure numerical way. In this case study both piecewise constant and piecewise linear feed rate profiles have been optimized with respect to the cost function (5). The durations of the time intervals are considered as additional free control parameters. The problem is first implemented with the gOPT *deterministic dynamic optimization code* [4]. This procedure shows slow convergence and a considerable sensitivity to the initialization of the control input. Therefore, also *stochastic and hybrid (deterministic-stochastic) programming codes* are evaluated [1]. The Hybrid Method for Dynamic Optimization (HyMDO) combines the key elements of a stochastic and a deterministic method, taking advantage of their complementary features. Firstly, the stochastic method ICRS/DS (Integrated Controlled Random Search for Dynamic Systems) is used to locate the vicinity of the global solution [1]. Secondly, this information is used to initialize gOPT. In other applications, such as, optimization of chemical reactors, the hybrid method proves to be superior –computationally and with respect to the cost value– to the purely deterministic or purely stochastic optimization. However, in this case study, the hybrid method shows no considerable benefit compared to the stochastic optimization method ICRS/DS. Even when starting at a solution which is quasi-optimal, the CPU-time for the gOPT optimization is unexpectedly high. As with the heuristic approach, different optimal control solutions are obtained depending on the imposed constraints, and this by all three optimization methods.

Simulation results. Three (of the multiple) optimal control solutions –with the corresponding cost (5) equal to 1– obtained by the heuristic and by the purely numerical approach are depicted in Figure 1. 1. The *heuristic profile* has two degrees of freedom for optimization with respect to the cost (5), namely, (*i*) the initial substrate concentration $C_S(0)$ (optimal value equal to 38.40 g/L), and (*ii*) the setpoint

value for the substrate concentration in a final feeding phase (optimal value equal to 0.95 g/L). When the heuristic profile is applied the experiment is finished when the maximum volume V_{MAX} is reached.
2. The graphically represented *purely numerical results* are obtained by the *stochastic* method (ICRS/DS) under the assumptions that (i) the initial concentration $C_S(0)$ and the final time t_f are fixed at the value of the optimal heuristically constructed feed rate profile (see above), and (ii) the final volume $V(t_f)$ is free but not greater than its maximum value V_{MAX}.

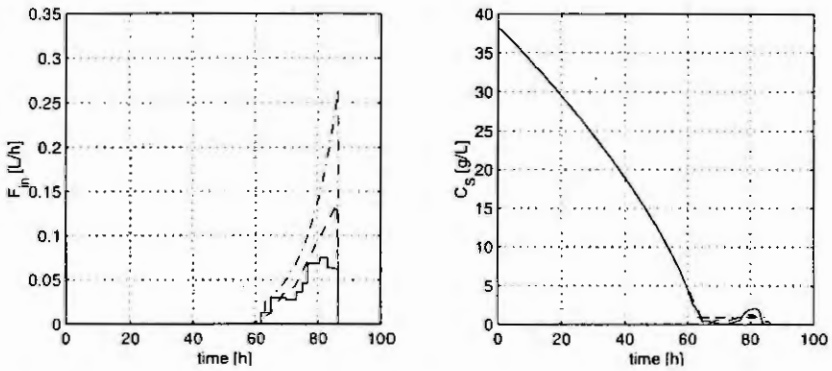


Figure 1: Feed rate (left plot) and substrate concentration (right plot) for an optimal heuristic solution (dashed-dotted line) and two optimal numerical solutions (dashed line: 8 ramps, full line: 16 steps).

Conclusions

In this paper we compared several strategies for solving the optimal control problem aimed at uncorrelated parameter estimation. The heuristic approach has been elaborated in previous work [5]. For the numerical approach, a stochastic program code is most suitable as it converges fast to an optimal solution, independently of the initially chosen input profile. This study points out that when using simple basis functions (without prior knowledge on the structure of the solution) also *multiple* optimal control solutions are obtained, i.e., similar as with the heuristic approach, a *multimodality* of the optimal control solution with respect to the modified E-design cost (5) is observed. Further, the purely numerical solutions are very similar to the heuristically obtained solutions when imposing well-specified constraints (Figure 1). On the one hand, the multimodality may be the reason (apart from the non-smoothness of the problem) for the slow (or non-) convergence of the deterministic optimization algorithm. On the other, it allows to select one unique identification experiment out of the available optimal control solutions by taking into account other features as model validity and practical feasibility.

Acknowledgment. Authors Karina Versyck and Els Dens are research assistants with the Fund for Scientific Research - Flanders (FWO). Work supported in part by Projects OT/95/20 and OT/99/24 of the Research Council of the Katholieke Universiteit Leuven, and the Belgian Programme on Interuniversity Poles of Attraction, initiated by the Belgian State, Prime Minister's Office for Science, Technology and Culture. The Fund for Scientific Research - Flanders (FWO) is acknowledged for the study journey grant that allowed to perform part of this research at the Instituto de Investigaciónes Mariñas (C.S.I.C.) in Vigo. The Centre for Process Systems Engineering (Imperial College, London) is acknowledged for an academic license of gOPT 1.5/gPROMS 1.4F. The scientific responsibility is assumed by its authors.

References

1. Banga, J. R., Alonso, A. A., and Singh, R. P., Stochastic dynamic optimization of batch and semi-continuous bioprocesses. Biotechnol. Progr., 13(3) (1997), 326 - 335.

2. Bryson, A. E. and Ho, Y., Applied Optimal Control. Hemisphere, New York, 1975.

3. Munack, A., Optimization of sampling. In: Biotechnology (Measuring, Modelling and Control), (Eds.: Rehm, H.-J. and Reed, G.) VCH, Weinheim, 4 (1991), 252 - 264.

4. Vassiliadis, V. S., Sargent, R. W. H. and Pantelides, C. C., Solution of a class of multistage dynamic optimization problems. Part I & II - Problems without path constraints. Ind. Eng. Chem. Res., 33 (1994), 2111 - 2122 and 2123 - 2133.

5. Versyck, K., Claes, J. and Van Impe, J., Practical identification of unstructured growth kinetics by application of optimal experimental design. Biotechnol. Progr., 13(5) (1997), 524 - 531.

NONLINEAR MODEL REDUCTION OF BIOPROCESS MODELS THROUGH SINGULAR PERTURBATION: AN ANALYTICAL SCALING APPROACH

S.R. Weijers and H.A. Preisig
Systems & Control group, Faculty of Applied Physics, Eindhoven University of Technology
Cascade 3.18, P.O. Box 513, NL-5600 MB Eindhoven, The Netherlands
Email: S.R.Weijers@tue.nl

Abstract. This paper focuses on nonlinear model reduction by timescale separation through singular perturbation to obtain slow and fast reduced order models. We used a model system with one biomass species and one substrate species with Monod kinetics to study reduction of continuous bioprocess systems. Applying an analytical scaling procedure was successful to bring the model system into the required so-called standard form for several operating conditions and subsequently derive reduced models for the fast and slow time scales. The results provide a mathematically and physically sound basis for order reduction of the continuous bioprocess model studied and a starting point for further model reduction studies through singular perturbation.

Introduction

Reduced order models are required for several important tasks. In control, they allow for reduced controller complexity and reduced computational requirements. Singular perturbation is a reduction approach based on timescale separation into fast and slow states and is especially suited for reduction of systems exhibiting multiple timescales. Moreover, singular perturbation theory provides the formal basis for application of quasi steady state assumptions that are a common tool to obtain reduced order models through neglect of fast dynamics.

A bioprocess model of a chemostat with one biomass species and one substrate species with Monod kinetics is used as a model system. This system was reduced with singular perturbation in [1], however without a firm motivation. The study in this paper provides such a firm motivation. For model reduction by singular perturbation, the model must be brought into the so-called standard form. This is usually the most difficult part in the reduction, especially for nonlinear systems, and this is the focus of this paper. The model system studied is brought into standard form for several operating conditions, and reduced order models are derived. To obtain this result in a systematic way, an analytical scaling procedure is applied that was suggested in [3].

The paper is organised as follows. The theory section introduces singular perturbation theory and the scaling procedure. Then the model system is described, followed by results and conclusions.

Singular perturbation theory and analytical scaling procedure

Let the system be described by n+m equations in state-space notation

$$\dot{x} = f(x,z,u,t,\varepsilon),\ x(t_0) = x_0, x \in R^n,\ u \in R^p \tag{1a}$$

$$\varepsilon\dot{z} = g(x,z,u,t,\varepsilon),\ z(t_0) = z_0, z \in R^m. \tag{1b}$$

with $\varepsilon>0$ a small scalar, the so-called perturbation parameter. Then for $\varepsilon\to 0$ the order reduces to n, because substituting a root $\bar{z}_i = \phi_i(\bar{x},\bar{u},t)$ of the equation $0 = g(x,z,u,t,0)$ in (1b) yields a reduced model:

$$\dot{\bar{x}} = f(\bar{x},\phi_i(\bar{x},\bar{u},t),\bar{u},t,0) \equiv \bar{f}(\bar{x},\bar{u},t),\ \bar{x}(t_0) = x_0, \tag{2}$$

which describes the slow dynamics of the system, also referred to as the outer system, or quasi steady state. Model (1) is in the so-called standard form if and only if in a domain of interest, the equation $0 = g(x,z,u,t,0)$ has $k\geq 1$ distinct real roots $\bar{z}_i = \phi_i(\bar{x},\bar{u},t), i = 1,2,..,k$. The quasi steady state $\bar{x}(t)$ is a uniform approximation of $x(t)$, that is, $\bar{x}(t) = x(t) + O(\varepsilon)$ holds for all $t\in[t_0, t_e]$, including t_0 as it can start from x_0. The quasi steady state $\bar{z}(t)$ however is not free to start from z_0, and the approximation $\bar{z}(t) = z(t) + O(\varepsilon)$ can be expected to hold only on an interval $t\in[\delta, t_e]$, with $\delta>t_0$. During an initial interval $[t_0,\delta]$ (the so-called boundary layer, or pre-steady-state), the original variable z approaches $\bar{z}$. The substitution $t_f = t/\varepsilon$ ("stretching" the initial time) converts (1) into a set of equations describing the fast dynamics of the system, the boundary layer or inner system (3).

$$\frac{d\hat{z}}{dt_f} = g(x_0,\hat{z}(t_f + \bar{z}(t_0),0,t_0),\hat{z}(0) = z_0 - \bar{z}(t_0) \tag{3}$$

The solution to this problem provides a boundary layer correction term $\hat{z} = \bar{z} - z$ which is used in a possible approximation $z = \bar{z}(t) + \hat{z}(t_f) + O(\varepsilon)$, valid for $t \in [t_0, t_s]$. Thus, singular perturbation theory allows us to treat slow and fast dynamics separately with reduced models for both timescales.

If a model to be reduced exhibits multiple timescales and is in the standard form, it is relatively straightforward to apply the order reduction as described above. Commonly, however, models are in non-standard form and in those cases, before the reduction can be applied they must be brought into standard form, if this is possible. A useful strategy to bring the model into standard form is to search for a (perturbation) parameter that is small relative to the other parameters. Typically, this is done through scaling of the equations. [2] gives scaling examples, including using dimensionless parameter groups, parameter scaling and state scaling. However, no systematic procedure is provided, and it was concluded that scaling requires considerable *a priori* knowledge. [3] proposes a more systematic procedure, based on analytic scaling which was applied successfully to analyse and test the Quasi Steady State Assumption (QSSA) that underlies Michaelis-Menten kinetics. The procedure is based on nondimensionalization of the model equations and is outlined as follows.

1. Derive analytical estimates for slow and fast timescales τ_f and τ_s.
2. Test on the necessary condition for timescale multiplicity, namely that $\tau_f \ll \tau_s$ holds.
3. Test on the necessary condition that the error in the slow state during the pre-steady-state is small.
4. For both timescales, choose an appropriate state scaling and derive dimensionless, scaled equations; this step should yield the standard form and the corresponding perturbation parameter.
5. Reduce the model.

The analytic scaling procedure is applied to the model system described in the next section.

Model system

A simple chemostat with one biomass and one substrate is used to test the analytic scaling procedure. After the description of the system, some analytical relationships are presented and different cases that are discussed in the next sections are indicated.

In a completely stirred tank reactor, biomass X grows on a single substrate S (the arrow in the formula denotes an autocatalytic reaction).

The reactor is shown schematically in Figure 1.

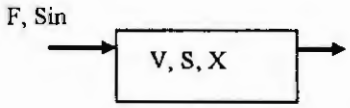


Figure 1: Chemostat with biomass growth on one substrate

For this system, assuming Monod kinetics the model equations write:

$$\dot{X} = \mu \frac{S}{K+S} X - D\,X, \quad \dot{S} = -k_1\, \mu \frac{S}{K+S} X - D\,S + D\,S_{in} \tag{4}$$

The dimensionless parameters defined for this system are the dimensionless residence time τ^* and the Monod number, Mo. For the system to be viable, no washout must occur so there is a lower limit to the dimensionless residence time (5).

$$\tau^* = \frac{\mu}{D}, \quad Mo = \frac{K}{S_{in}}, \quad \tau^*_{min} = \frac{\mu}{D_{max}} = 1 + Mo \tag{5}$$

The steady state concentrations and the ratio S/X in steady state written as a function of the dimensionless parameters τ^* and Mo are:

$$X_\infty = \frac{1}{k_1}\{S_{in} - \frac{DK}{(\mu - D)}\}, \qquad S_\infty = \frac{DK}{(\mu - D)}, \quad \frac{S_\infty}{X_\infty} = \frac{1}{k_1}\{\frac{Mo}{\tau^* - 1 - Mo}\} \tag{6}$$

Upon linearization in a state $x=\{X, S\}$ with $u=D$, the linearized system (7) is obtained with eigenvalues (8).

$$\dot{x} = \begin{bmatrix} \frac{\mu S}{K+S} - D & \frac{\mu K X}{(K+S)^2} \\ -k1\frac{\mu S}{K+S} & -k1\frac{\mu K X}{(K+S)^2} - D \end{bmatrix} x + \begin{bmatrix} 0 \\ S_{in} \end{bmatrix} u \qquad (7) \qquad \lambda_{1,2} = -D,\ -D + \mu\frac{S(S+K) - k_1 XK}{(S+K)^2} \qquad (8)$$

Starting point in the subsequent timescale analysis is the conjecture that occurrence of multiple timescales is associated with a large concentration difference between the state variables, the fast dynamics being associated with the state variable with the smallest value. Three cases are distinguished as indicated in Table 1 and briefly explained below.

Table 1: Cases distinguished in timescale analysis; for explanation, see text

Case	Ratio S_∞/X_∞	Condition for τ^*	Other condition
1	$S << X$	$\tau^* >> 1$	$Mo << 1$
2	$S << X$	$\tau^* >> 1$	$Mo \approx 1$
3	$S << X$	$\tau^* > O(2)$[1]	$Mo << 1$

[1]: "Big O" stands for "order of magnitude"; for a formal definition see [2]

From (6), we see that S/X is low when Mo is very small or τ^* very large. Case 1 and 2: When τ^* is very large, substrate conversion is almost complete and biomass produced is only slowly withdrawn. At high feed substrate concentration S_{in} (small Monod, Case 1), the concentration difference is larger at a given residence time than at lower S_{in}, because the biomass concentration is higher whilst the substrate concentration remains unchanged (Case 2). Case 3: At moderate dilution rate, but far from washout, which is expressed by the condition $\tau^* > O(2)$, the ratio S/X can still be small when the feed concentration is very high (small Monod). For typical values representing the different cases, the ratio S_∞/X_∞ and the ratio of the eigenvalues are given in Table 2. Equations (6) and (8) were used, with the following parameter values: μ=4, K=20, k_1=1.5.

Table 2: Ratio S_∞/X_∞ and ratio of eigenvalues for different cases

Case	τ^*	Mo	S_∞/X_∞	λ_1/λ_2	λ_1	λ_2
1	20	0.04	0.0032	0.0022	0.2	90
2	20	0.4	0.0323	0.0226	0.2	8.8
3	3	0.04	0.0306	0.0306	1.33	43.5

The results indicate that the supposed relationship between the ratio S_∞/X_∞ and the ratio of eigenvalues holds indeed. In the next sections, the cases are analysed according to Table 1.

Results

First Case 1 and Case 2 are considered. Scaling is performed employing estimates based on the analytical solutions of the eigenvalues of the system, which leads to the standard form.

1. Estimate timescales. In Case 1 and 2, the large eigenvalue of (8) can be approximated as follows, because $X>>S$, $S<<K$, $\mu>>D$ (because $\tau^*>>1$) and $X \approx S_{in}/k_1$:

$$\lambda_2 = -D + \mu\frac{-k_1 XK + S(S+K)}{(S+K)^2} \approx -D - \mu\frac{k_1 XK}{(S+K)^2} \approx -D - \frac{\mu}{Mo} \approx -\frac{\mu}{Mo} \qquad (9)$$

The fast timescale and the slow timescale are scaled with the time constants, which are reciprocal to the real parts of the eigenvalues.

$$t_f = t\cdot\mu/Mo\,, \quad t_s = t\cdot D \qquad (10)$$

2. Test on timescale multiplicity. For the QSSA to be valid, $\tau_f << \tau_s$ must hold, so we have condition (11), which holds under the assumptions made

$$Mo/\mu << 1/D, \qquad \text{or} \quad \tau^*/Mo >> 1 \qquad (11)$$

3. Test on smallness of error on the initial condition for the slow state during the pre-steady-state. The relative error in the slow state is approximated with (12).

$$\left|\frac{\Delta x}{x_0}\right| \approx \frac{1}{x_0}\left|\frac{dx}{dt}\right|_{max} \cdot \tau_f \tag{12}$$

For the model system, we have τ_f=Mo/μ for the fast time scale. Assuming that S_0 and X_0 are in the same order of magnitude as S_∞ and X_∞ respectively the error is approximated by the estimate (13), which is small indeed.

$$\left|\frac{\Delta X}{X_0}\right| \approx \left|\mu\frac{S_0}{K+S_0} - D\right| \cdot \frac{Mo}{\mu} \approx \left|\mu\frac{S_0}{K} - D\right| \cdot \frac{Mo}{\mu} = \left|\frac{S_0}{S_{in}} - \frac{Mo}{\tau}\right| = O(\varepsilon) \tag{13}$$

4. For timescales, choose state scaling and derive scaled equations and (try to) find perturbation parameter. The states are scaled with their approximate steady state values, so that the resulting dimensionless state variables are both O(1). Substitution of scaled variables $x=X\cdot k_1/S_{in}$ and $s=S\cdot\tau^*/K$ and for the slow time scale $t_s= t\cdot D$ yields:

$$D\frac{dx}{dt_s} = \mu\frac{sK/\tau^*}{K+sK/\tau^*}x - Dx \tag{14a}$$

$$D\frac{dsK/\tau^*}{dt_s} = -k_1\mu\frac{sK/\tau^*}{K+sK/\tau^*}x\frac{S_{in}}{k_1} - Ds\frac{K}{\tau^*} + DS_{in} \tag{14b}$$

Dividing (14a) and (14b) by D, multiplying numerator and denominator of the Monod term by K, using Mo=K/S_{in} and introducing $1/\tau^*$ as a perturbation parameter yields the outer equations (15) which are easily seen to be in standard form.

$$\frac{dx}{dt_s} = \frac{s}{1+\varepsilon s}x - x \tag{15a}$$

$$\varepsilon\frac{ds}{dt_s} = -\frac{s}{1+\varepsilon s^*}x\frac{1}{Mo} - \varepsilon s + \frac{1}{Mo} \tag{15b}$$

The substitution $t_f= t\cdot\mu$/Mo gives the inner equations (16).

$$\frac{dx}{dt_f} = \varepsilon Mo\left(\frac{s}{1+\varepsilon s}x - x\right) \tag{16a}$$

$$\frac{ds}{dt_f} = -\frac{s}{1+\varepsilon s^*}x - \varepsilon Mo\ s + 1 \tag{16b}$$

Thus the timescale estimation and knowledge of scaling of the variables enables a scaling of the variables which in turn has led to successful selection of a perturbation parameter $1/\tau^*$, as $\tau^*>>1$, so that $\varepsilon<<1$.

In Case 3, scaling also leads to the standard form and to selection of a perturbation parameter. Here, S<<K no longer holds, because S=O(K) and s=S/K is used for scaling. The standard form is obtained after some manipulations, with Mo^* as a perturbation parameter.

Conclusions

A systematic scaling procedure was successfully applied to transform a continuous bioprocess model into the so-called standard form, which is required for application of singular perturbations to obtain slow and fast model reduced order models. The results provide a mathematically and physically sound basis to apply model order reduction through singular perturbation to the model studied. The scaling procedure constitutes a starting point for further model reduction studies of bioprocess model and other models by singular perturbations.

References

1. Bastin, G. and D. Dochain (1990) *On-line Estimation and Adaptive Control of Bioreactors*, Process Measurement and Control, 1, Elsevier, Amsterdam.
2. Kokotovic, P, H. Khalil and J. O'Reilly (1986) *Singular perturbation methods in control: Analysis and design*, Academic Press, London.
3. Segel, L.A. and M. Slemrod (1989) *The quasi-steady state assumption: a case study in perturbation*. SIAM Review Vol. 31 (3) pp. 446-477.

STOCHASTIC PERTURBATION ANALYSIS OF A MICROBIAL GROWTH MODEL

N. Scheerlinck [1], F. Poschet [2], J. F. Van Impe [2] & B. M. Nicolaï [1]

[1] Laboratory for Postharvest Technology, Katholieke Universiteit Leuven,
W. de Croylaan 42, B–3001 Heverlee, Belgium
E–mail: Nico.Scheerlinck@agr.kuleuven.ac.be
[2] BioTeC – Bioprocess Technology and Control, Katholieke Universiteit Leuven,
Kardinaal Mercierlaan 92, B–3001 Heverlee, Belgium

Abstract
An algorithm has been developed to analyse the propagation of uncertainties during microbial growth processes. Variability on the model parameters was incorporated in the growth model of Baranyi, and a perturbation algorithm for the computation of the mean and variance of the microbial load has been developed and implemented. The algorithm is based on the computation of the propagation of an infinitesimal perturbation on the (stochastic) growth parameters. Mean values and variances of the microbial load are then easily evaluated. The efficiency of the perturbation algorithm has been compared to that of a Monte Carlo algorithm. It was shown that the former is much faster than the latter. Also, the results obtained with both algorithms are comparable. Simulations indicate that stochastic fluctuations of the growth parameters may cause a considerable level of uncertainty in the microbial load.

Introduction

The objective in food processing is to reduce microbial spoilage and to ensure good quality and microbiological safety of end products towards the consumers. In order to quantify the risk of a consumer's infection, until now most of the microbial tests are post factum analyses. As measurements are tedious and time–consuming, it is important to be able to rely on appropriate microbial growth simulation tools for an accurate prediction of the transient course of the microbial load inside a food product during the whole production process. Most of these tools are mainly based on deterministic mathematical models which are developed in the field of predictive microbiology [4]. It is assumed that the growth parameters and the initial and process conditions are accurately known. The predicted microbial load at a certain time instance is, therefore, fully deterministic. Unfortunately, because of their biological nature, the model parameters may vary considerably between and within different product samples. In real operation mode also the process conditions are subjected to unpredictable random fluctuations. As a consequence, the microbial load is also random and should be considered as a stochastic variable characterised by its statistical characteristics such as the mean value, variance and probability density function. This results in a variable product safety, and possibly end products with microbial loads which are beyond the acceptable threshold. A proper food process design and risk analysis should take this variability into account [2], [4]. The objective of this paper was to investigate the effect of random parameter uncertainties on the predicted microbial load based on the growth model of Baranyi [1] by means of a perturbation algorithm.

Incorporation of uncertainty in the growth model of Baranyi

According to Baranyi [1], microbial growth under dynamic environmental conditions can be described by means of the following nonlinear system of ordinary differential equations

$$\frac{d}{dt}\mathbf{y}(t,\mathbf{p}) = \mathbf{f}(\mathbf{y}(t,\mathbf{p}),\mathbf{p}) \quad (1)$$

$$\mathbf{y}(t_0,\mathbf{p}) = \mathbf{y}_0(\mathbf{p}) \quad (2)$$

where

$$\mathbf{y} = \begin{bmatrix} n & q \end{bmatrix}^T \qquad \mathbf{y}_0 = \begin{bmatrix} n_0 & q_0 \end{bmatrix}^T$$
$$\mathbf{f} = \begin{bmatrix} \mu_{max}\dfrac{1-\exp(n-n_{max})}{1+\exp(-q)} & \mu_{max} \end{bmatrix}^T \qquad q_0 = \ln\left[\exp(\lambda\,\mu_{max})-1\right]^{-1} \quad (3)$$
$$\mathbf{p} = \begin{bmatrix} n_0 & n_{max} & \mu_{max} & \lambda \end{bmatrix}^T$$

with n the Neperian logarithm of the microbial load [ln(CFU/mL)], q the Neperian logarithm of the intracellular physiological state [-], n_0 the Neperian logarithm of the initial microbial load [ln(CFU/mL)], q_0 the Neperian

logarithm of the initial intracellular physiological state [-], n_{max} the Neperian logarithm of the maximal microbial load [ln(CFU/mL)], μ_{max} the maximum specific growth rate [1/h], λ the lag phase [h], t the time [h] and t_0 the initial time [h]. For the research conducted in this paper, the microbial growth will be considered under static environmental conditions, e.g., the process conditions are considered to be constant (temperature, pH, ...).

The model of Baranyi is deterministic in the sense that the transient course of the microbial load is entirely predetermined by the initial growth conditions (n_0, q_0) and the exact knowledge of the model parameter set $\mathbf{p}$. Unfortunately, in food microbiology practice, it is known that the initial conditions can vary considerably [2]. Furthermore, biological variability inherently causes variability on the estimated model parameters $\mathbf{p}$. The concept of random variables can be used to introduce this variability in the Baranyi model. It is assumed here that the parameter set $\mathbf{p}$ consist of random variables characterised by a multivariate probability function $\mathbf{f_p}$ with mean $\bar{\mathbf{p}}$ and covariance matrix $\mathbf{V_{p,p}}$.

Monte Carlo Method

Because of the uncertainty on the Baranyi growth model parameters, the model outputs will be uncertain as well and should be considered as stochastic variables together with their statistical characteristics such as the mean value, variance and probability density function.

By measuring the microbial load of specific organisms in a large number of product samples under static conditions and by performing some statistical analyses on the measurements, the transient course of the microbial load can be tracked and described in a statistical way. This obviously represents a considerable experimental effort. A straightforward statistical approach to the solution of microbial growth models with random parameters is the Monte Carlo method. In this method a sample of the random parameter set $\mathbf{p}$ is generated on the computer and the corresponding growth model is numerically solved. This procedure is repeated several times and finally the mean values and variances, but also higher order moments, can be estimated using common statistical techniques.

Major drawbacks of the Monte Carlo method are the large number of repetitive simulations necessary to obtain an acceptable level of accuracy and the fact that the stochastic parameter set $\mathbf{p}$ must be completely specified in a probabilistic sense.

Perturbation Method

An alternative algorithm for the computations of the mean values and variances of the microbial load is based on a probabilistic perturbation scheme of the Baranyi growth model (1-2). Hereto the state vector $\mathbf{y}$ and system function $\mathbf{f}$ are expanded into a first order Taylor series

$$\mathbf{y}(t, \mathbf{p}) = \mathbf{y}(t, \bar{\mathbf{p}}) + \left[\frac{\partial}{\partial \mathbf{p}}\mathbf{y}(t, \bar{\mathbf{p}})\right]\Delta\mathbf{p} \tag{4}$$

$$\mathbf{f}(\mathbf{y}, \mathbf{p}) = \mathbf{f}(\mathbf{y}(t, \bar{\mathbf{p}}), \bar{\mathbf{p}}) + \left[\frac{\partial}{\partial \mathbf{y}}\mathbf{f}(\mathbf{y}(t, \bar{\mathbf{p}}), \bar{\mathbf{p}})\right]\left[\frac{\partial}{\partial \mathbf{p}}\mathbf{y}(t, \bar{\mathbf{p}})\right]\Delta\mathbf{p} + \left[\frac{\partial}{\partial \mathbf{p}}\mathbf{f}(\mathbf{y}(t, \bar{\mathbf{p}}), \bar{\mathbf{p}})\right]\Delta\mathbf{p} \tag{5}$$

where the Jacobians in Eqs (4-5) are evaluated using the mean parameter set $\mathbf{p} = \bar{\mathbf{p}}$. Substitution of Eqs (4-5) in Eq (1-2) and re-ordening terms results in the following systems

$$\frac{d}{dt}\mathbf{y}(t, \bar{\mathbf{p}}) = \mathbf{f}(\mathbf{y}(t, \bar{\mathbf{p}}), \bar{\mathbf{p}}) \tag{6}$$

$$\frac{d}{dt}\left[\frac{\partial}{\partial \mathbf{p}}\mathbf{y}(t, \bar{\mathbf{p}})\right] = \left[\frac{\partial}{\partial \mathbf{y}}\mathbf{f}(\mathbf{y}(t, \bar{\mathbf{p}}), \bar{\mathbf{p}})\right]\left[\frac{\partial}{\partial \mathbf{p}}\mathbf{y}(t, \bar{\mathbf{p}})\right] + \left[\frac{\partial}{\partial \mathbf{p}}\mathbf{f}(\mathbf{y}(t, \bar{\mathbf{p}}), \bar{\mathbf{p}})\right] \tag{7}$$

with initial conditions

$$\mathbf{y}(t_0, \bar{\mathbf{p}}) = \mathbf{y}_0(\bar{\mathbf{p}}) \tag{8}$$

$$\left[\frac{\partial}{\partial \mathbf{p}}\mathbf{y}(t_0, \bar{\mathbf{p}})\right] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & \dfrac{\partial q_0}{\partial \mu_{max}} & \dfrac{\partial q_0}{\partial \lambda} \end{bmatrix} \tag{9}$$

Applying the mean value operator $\mathbf{E}$ on Eq (4) yields

$$\bar{\mathbf{y}} = \mathbf{E}\left[\mathbf{y}(t, \mathbf{p})\right] = \mathbf{y}(t, \bar{\mathbf{p}}) \tag{10}$$

which means that a first order approximation of the mean microbial load and the physiological state can be obtained by solving the deterministic Baranyi growth model using the mean value parameter set $\mathbf{p} = \overline{\mathbf{p}}$.

The covariance matrix at an arbitrary time t can be computed as follows

$$\mathbf{V}_{\mathbf{y},\mathbf{y}}(t) = \mathrm{E}\left[(\mathbf{y} - \overline{\mathbf{y}})(\mathbf{y} - \overline{\mathbf{y}})^T\right] = \left[\frac{\partial}{\partial \mathbf{p}}\mathbf{y}(t, \overline{\mathbf{p}})\right] \mathbf{V}_{\mathbf{p},\mathbf{p}} \left[\frac{\partial}{\partial \mathbf{p}}\mathbf{y}(t, \overline{\mathbf{p}})\right]^T \tag{11}$$

Application to the microbial growth of *E. coli K12*

The developed perturbation algorithm was implemented in MATLAB (The MathWorks Inc., Natick, Massachusetts) and applied to compute the uncertainty propagation under static conditions for the microbial growth of *E. coli K12*.

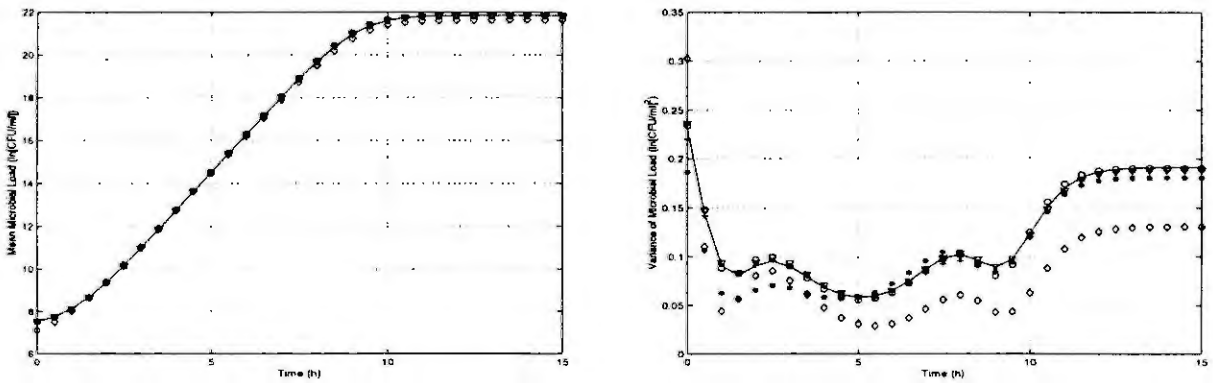


Figure 1: Mean and variance of the microbial load for *E. coli K12*. line: Monte Carlo 100000 runs; o: First order perturbation method; ∇: Monte Carlo 10000 runs; +: Monte Carlo 1000 runs; *: Monte Carlo 100 runs; ◇: Monte Carlo 10 runs.

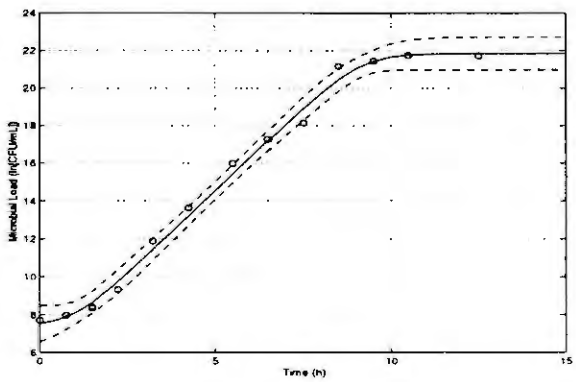


Figure 2: Confidence intervals of the microbial load for *E. coli K12*. o : data points obtained by measurements; mean microbial load $\overline{n}$ (line) and $\overline{n} \pm 2\sigma_n$ (–) computed by means of the perturbation algorithm

The results obtained by means of the first order perturbation method were compared with these obtained by the Monte Carlo method. The model parameters were considered to be normal distributed from which the mean values $\overline{\mathbf{p}}$ and the covariance matrix $\mathbf{V}_{\mathbf{p},\mathbf{p}}$ were extracted from previous research [3].

$$\overline{\mathbf{p}} = \begin{bmatrix} 7.52 & 21.85 & 1.78 & 1.05 \end{bmatrix}^T \; ; \; \mathbf{V}_{\mathbf{p},\mathbf{p}} = \begin{bmatrix} 0.23 & -0.0021 & 0.0093 & 0.16 \\ -0.0021 & 0.19 & -0.0072 & -0.0110 \\ 0.0093 & -0.0072 & 0.0097 & 0.027 \\ 0.16 & -0.0110 & 0.027 & 0.1763 \end{bmatrix} \tag{12}$$

In Figure 1 the time course of the mean and variance of the microbial load are shown. For the mean transient course of the microbial load, there is a good agreement between the results obtained with the perturbation method

and these obtained with several Monte Carlo methods. For the computation of the variance of the microbial load, there is a good agreement between the perturbation method and Monte Carlo methods with large number of runs. This indicates that a large number of Monte Carlo runs is required to obtain sufficiently accurate results, especially when the computation of variances is considered. It is clear that the variance of the microbial load changes with time and that it remains significant during the whole growth process. The variance related to the initial conditions fades out and after some time the effect of the variance of the other random parameters becomes apparent. This may be important for food process design because the uncertainty of the microbial growth parameters can cause a considerable level of uncertainty on the effect of the food process and the safety of end products. In Figure 2 the measurement data points and the $(\overline{n} \pm 2\sigma_n)$ intervals for correlated growth parameters are shown. The intervals can be considered as a sort of 95% confidence intervals. In Table 1 the timing results (CPU), the approximate number of floating point operations (FLOPS) and the relative CPU time (REL) for the different algorithms and stochastic models are summarized. The relative CPU time is defined as the required CPU time divided by the CPU time to solve the equivalent determinstic problem. The perturbation method executes clearly faster as compared to the Monte Carlo methods.

Table 1: CPU time (s) for the uncertainty propagation analysis of microbial growth processes with random variable model parameters. CPU: execution time; FLOPS: number of floating point operations; REL: relative CPU time.

Algorithm	CPU	FLOPS	REL	Algorithm	CPU	FLOPS	REL
First order perturbation	2	$2.7\ 10^5$	1.3	Monte Carlo 1000 runs	303	$3.2\ 10^7$	218
Monte Carlo 100000 runs	30309	$3.2\ 10^9$	21805	Monte Carlo 100 runs	30	$3.3\ 10^6$	21.8
Monte Carlo 10000 runs	3031	$3.2\ 10^8$	2180	Monte Carlo 10 runs	3	$3.1\ 10^5$	2.3

Conclusions

Food process performance in terms of safety end products is usually quoted or designed based on deterministic microbial growth models in which the model parameters need to be accurately known. Because of the inevitable random nature of the microbial growth process, variability on the microbial growth parameters was introduced in the model of Baranyi. A first order perturbation algorithm for microbial growth with stochastic model parameters has been proposed and implemented. The stochastic model parameters were considered to be of the random variable type. Based on simulations of growth of *E. coli K12* it was shown that the perturbation method is a powerful alternative to the Monte Carlo method. Also, random variable fluctuations of the growth parameters may cause considerable uncertainties in the time course of the microbial load inside processed food products.

Acknowledgements

The European Union (FAIR projects CT96–1192 and CT97-3129), the Flemish Minister of Science and Technology (COF97/08 and COF98/08) and the K.U.Leuven Research Council (OT/99/24) are gratefully acknowledged for their financial support. K. Bernaerts (BioTeC, Katholieke Universiteit Leuven) is kindly acknowledged for providing the data set of *E. coli K12*.

References

[1] J. Baranyi and A. Roberts. A dynamic approach to predicting bacterial growth in food. *International Journal of Food Microbiology*, 23:277–294, 1994.

[2] B.M. Nicolaï and J.F. Van Impe. Predictive Food Microbiology: a probabilistic approach. *Mathematics and Computers in Simulation*, 42:287–292, 1996.

[3] F. Poschet and J.F. Van Impe. Quantifying the uncertainty of model outputs in predictive microbiology: a Monte Carlo analysis. *Mededelingen Faculteit Landbouwwetenschappen Universiteit Gent* , 64((5b)):499–506, 1999.

[4] I. Walls and V.N. Scott. Use of Predictive Microbiology in Food Savety Risk Assessment. *International Journal of Food Microbiology*, 36:97–102, 1997.

SIMULATION OF A BIOPROCESS FOR OPERATORS' TRAINING

M.N. Pons, F. Parmentier, J.P. Corriou, M. Baklouti
Laboratoire des Sciences du Génie Chimique, CNRS-ENSIC-INPL
1, rue Grandville, BP 451, F-54001 Nancy cedex, France

Abstract. A complete bioprocess (baker's yeast production), including the control loops for temperature, pH and dissolved oxygen, has been modeled and simulated in three different environments (FORTRAN, Matlab/Simulink and WinSim) to offer a training platform for operators.

Introduction.

Bioprocesses are characterized by the use of living micro-organisms that require a well-balanced environment in terms of medium composition, pH, temperature, dissolved oxygen, etc. This is obtained by a set of control loops that are highly interconnected. It has also to be noticed that bioprocesses are essentially non-linear systems: this is due to the non linear nature of the metabolism as well as to the mode of operation (batch or fed-batch). In industry, most of the controllers are PIs, normally valid for linear systems. As a consequence of the specificity of bioprocesses constant optimal controller settings cannot be found for the entire duration of a fermentation run: they should be updated either manually by the operators or automatically, when some model of the bioprocess is available. It has to be acknowledged that this is not often the case in industry.

The mastering of the controls by operators necessitates a training that can be long when realized on real fermentations, as many of them last several days. To improve this training in terms of time and cost, a complete simulator of the bioprocess can be proposed.

The simulation of the process is based on the combination of various models, describing the hydrodynamics of the vessel, the (bio)chemical reactions, the thermodynamics and the control devices (sensors, valves, pumps, controllers). This contribution describes the development of a baker's yeast production simulator in a well-stirred reactor taking into consideration the different control loops.

Process description.

Baker's yeast is growing on a carbon substrate (sugar S) in presence of oxygen, of a nitrogen source and of different minerals (sources of P, S, Ca, Na, Mg, K, etc.). Simultaneously metabolites such as ethanol, carbon dioxide and acetic acid are produced. The behavior of this yeast with respect to carbon sources is complex: if sugar is in excess (≥ 0.1 g/l), the yeast will grow and produce mainly ethanol (E) and CO_2 (metabolic state X_1). If the sugar concentration lays in the range $0 - 0.1$ g/l, the yeast will essentially grow and produce CO_2 (metabolic state X_2). If no sugar is available, the yeast can consume ethanol in presence of oxygen (metabolic state X_3). Acetic acid (Ac), which is always produced in small amounts, is known to inhibit the growth above a certain concentration [1].

An optimal yeast production is obtained when the cell is maintained in the metabolic state X_2, as ethanol production is then minimized. Temperature and pH have to be maintained at optimal values: 32°C and 4.2 respectively. Dissolved oxygen should not be limiting. To optimize the productivity, the process is generally operated fedbatchwise, by addition of sugar and mineral salts. A typical process with a jacketed reactor and the different control loops is presented in Figure 1.

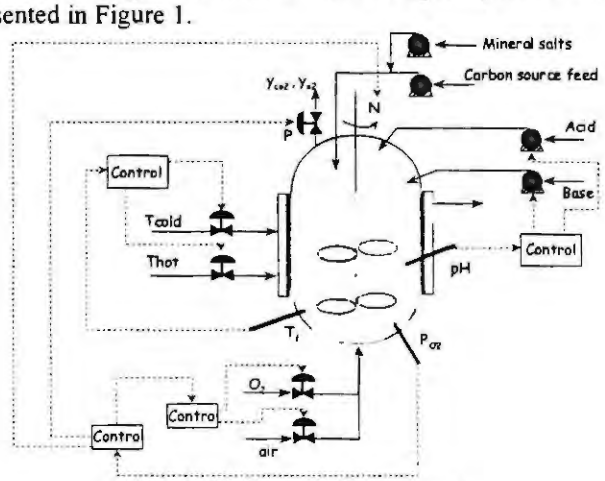


Figure 1: Fermentation process

Modelling.

Bioreactions. The model proposed by Rajab [2] is used. The cell will adapt its metabolism in function of the substrate concentrations and along its life can be found in any metabolic state. The change from one metabolic state to another is not instantaneous but is reversible. The total biomass (X) is the sum of the biomass in the three metabolic states: $X = X_1 + X_2 + X_3$. The growth is described for each metabolic state by the following equations (Monod's type) (for sake of simplicity, the limitation effects of mineral salts have been omitted here):

For X_1 : $r_{x1} = \mu_{max,1} \frac{S}{K_{g1} + S} \cdot \frac{1}{1 + Ac_T / K_{a1}} \cdot \frac{O_2}{K_{O2} + O_2} \cdot X_1$

For X_2: $r_{x2} = \mu_{max,2} \frac{S}{K_{g2} + S} \cdot \frac{1}{1 + Ac_T / K_{a2}} \cdot \frac{O_2}{K_{O2} + O_2} \cdot X_2$

For X_3 : $r_{x3} = \mu_{max,3} \left(\frac{E}{K_{e3} + E} + \varphi \frac{Ac_T}{K_{e3} + Ac_T} \right) \cdot \frac{1}{1 + Ac_T / K_{a3}} \cdot \frac{O_2}{K_{O2} + O_2} \cdot X_3$ with $\varphi = 1$ if $E \leq 10^{-6}$ g/l and $\varphi =$ 0 otherwise. The maximal growth rates ($\mu_{max,i}$) are function of pH and temperature. The transition rates are given by:

$$r_{2\to1} = k_{21} X_2 \frac{S}{S_{crit} + S}, \quad r_{1\to2} = k_{12} X_1 \left(1 - \frac{S}{S_{crit} + S} \right), \quad r_{3\to1} = k_{31} X_3 \frac{S}{K_{eq31} + S}, \quad r_{3\to2} = k_{32} X_3 \frac{S}{K_{eq32} + S}$$

$$r_{1\to3} = k_{13} X_1 \frac{E}{K_{eq13} + E} \cdot \left(1 - \frac{S}{S_{crit} + S} \right), \quad r_{2\to3} = k_{23} X_2 \frac{E}{K_{eq23} + E}$$

The production rates of metabolites and the consumption rates of substrates and mineral salts [3] are deduced from the growth rates by using constant yields.

Hydrodynamics. The reactor is supposed to be well mixed. The variation of volume due to the feeding of sugar, mineral salts and pH reagent is taken into consideration.

Heat transfer. The reactor temperature is controlled by a circulation of hot and/or cold water in the jacket or by the circulation of cold water heated by an electrical resistance. The heat balance takes into account the jacket and the reactor, as well as the effect of agitation (aerated power).

pH: The culture medium can contain various ions: NH_4^+, cations such as Ca^{2+} , Na^{2+}, K^+, anions such as chloride, sulfate, acetate, carbonate, bicarbonate, phosphate, etc.. The pH calculation method proposed by Pons et al. [4] has been adopted.

Gas phase: Mass balances are necessary to determine the molar fraction of oxygen and CO_2 in the off-gas as well as the dissolved oxygen and carbon dioxide concentrations. Various classical correlations are used to estimate the non-aerated and aerated powers and the oxygen transfer coefficient. The total pressure in the vessel is taken into account.

Controllers.

PI-type controllers with anti-windup have been selected. For temperature, the manipulated variable is either the percentage of opening of the hot water valve, the total water flow rate in the jacket being constant (hot / cold water case) or the electrical power when an electrical resistance is used (constant cold water flow rate). The dissolved oxygen control can be achieved by various combinations of the available manipulated variables: total gas flow rate, fraction of oxygen in the aeration gas, pressure, stirring speed. Different control schemes can also be proposed for pH: use of a unique reagent (base or acid depending upon the fermentation type) or of two reagents. In the case of baker's yeast, pH is naturally decreasing as nitrogen is incorporated into biomass for growth. pH is often controlled by addition of pulses of a solution of ammonia when pH is lower than a given set point. When two reagents are used, an on/off scheme with two set points or a dead-zone controller can be used.

Simulation environments.

Three different simulation environments have been used: FORTRAN-DVF, Matlab/Simulink and WinSim 3.0 (RSI, Montbonnot, France).

FORTRAN-Digital Visual Fortran 5.0 (Digital, Maynard, USA) was used by undergraduate students for a computer science project aimed at developing a bioprocess simulator. The main advantage here is to be able to program exactly the equations defined previously but it can be difficult to build user-friendly interfaces. Furthermore maintaining and improving the software are not easy tasks.

Matlab/Simulink offers a programming environment devoted to control studies. However, in the present case, the flowsheet can be complex due to the number of interconnected blocks (over 50). Some parts have been

programmed directly as S-functions. The plots of the different state variables with respect to time are easy to obtain.

WinSim 3.0 is a dedicated software for the simulation of chemical processes under Windows for operators' training. It contains a library of process units (distillation columns, etc.) and controllers. New process units can be defined by the user under C++. Two toolboxes are available: a "Process diagrams" toolbox to define the process (units, connections, pumps, valves, controllers, transmitters, alarms, etc) and a "Mimics" toolbox to build a man-machine interface and to control the simulated process. The interface (Figure 2) is very similar to the one the operator will have in a plant control room. However the addition of new units requires a good knowledge of C++.

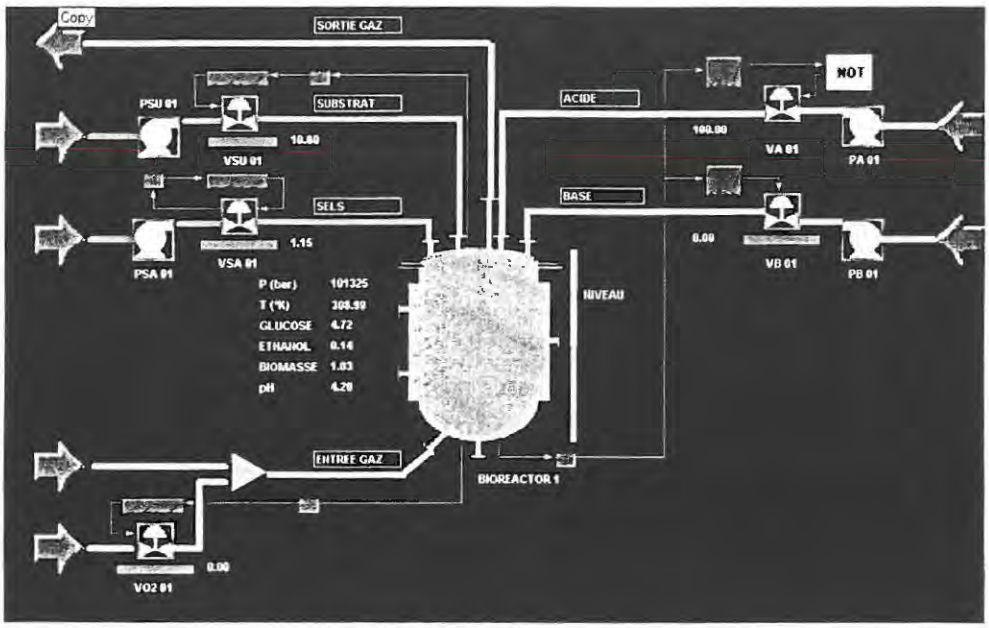


Figure 2: WinSim 3.0 operator interface

Results.

Figure 3 presents some curves obtained with the Matlab/Simulink environment for a fedbatch operation with a constant sugar feed rate of 0.1 l/h (concentration 300 g/l). The initial volume is 10 l. The dissolved oxygen is controlled at 30% saturation by manipulation of the stirring speed (in the range 350 – 500 rpm). The inlet gas flow rate increases linearly from 250 l/h to 500 l/h during the 10 hours of operation. At 5.5 hrs the maximal stirring speed is reached and the dissolved oxygen cannot be controlled anymore at the desired setpoint. pH is controlled at 4.2 by addition of pulses of an ammonia solution. During the first four hours, glucose is above the critical value of 0.1 g/l and the metabolic state X_1 is favored. Later, X_2 is favored. These results are very similar to those which can be obtained in a real experiment.

Conclusions.

A complete baker's yeast process, that includes all the control loops that can be found on an actual fermentor, has been modeled and simulated using three programming environments. The size of the problem makes Matlab/Simulink somewhat cumbersome to use and the interface is not visual enough for operators' training. It may be however used in classroom with students. WinSim is more adapted to operator training as the operator can modify the settings (controllers, flow rates, setpoints) as on a real process and visualize the actual effects of his/her actions, but it requires a good knowledge of C++ for development.

Work is going on to further develop the WinSim platform to facilitate the choice of control strategies and simulate large-scale fermentors.

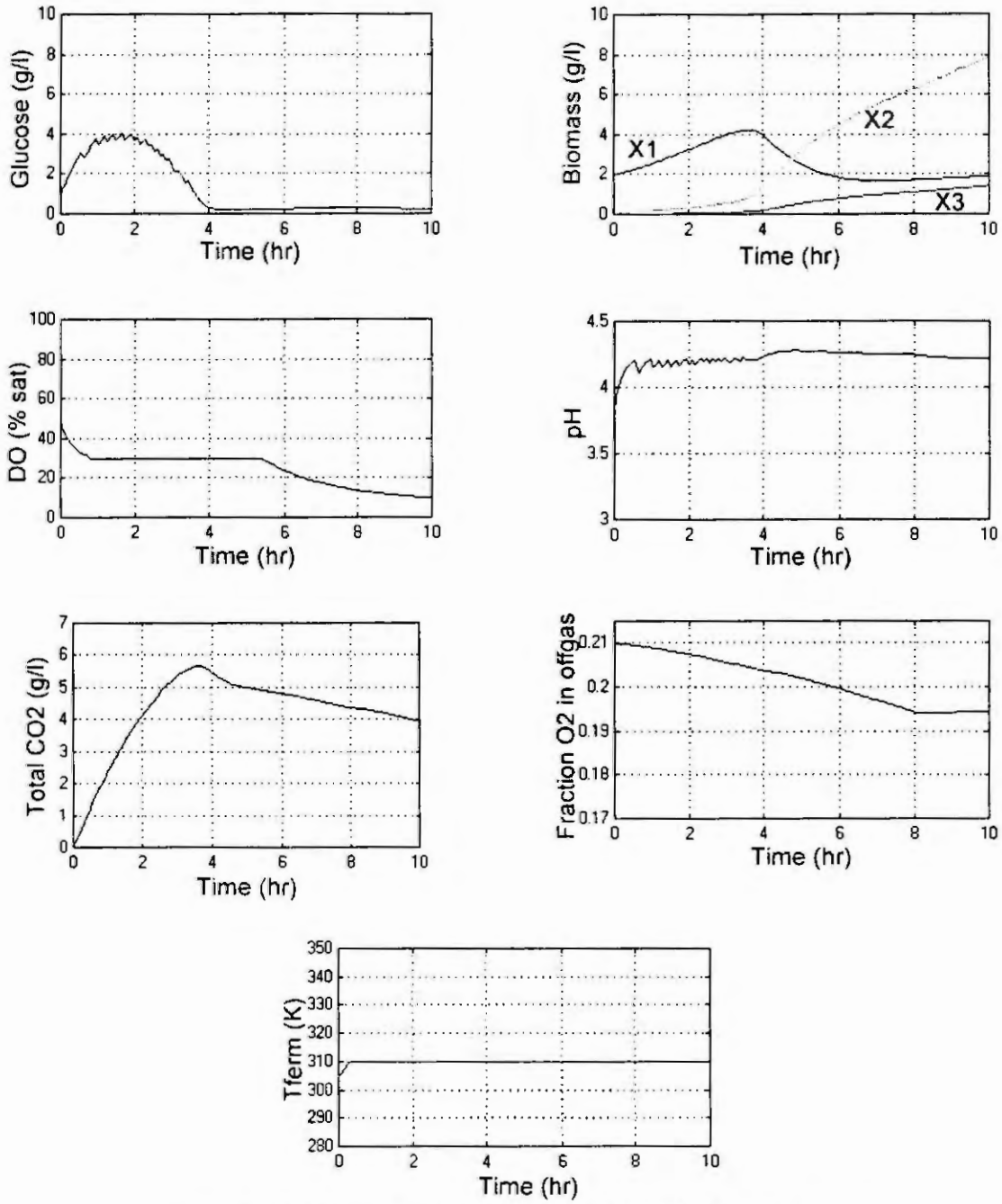


Figure 3: Some of the plots obtained with the Matlab/Simulink environment

References.

1. Pons, M.N., Rajab, A. and Engasser, J.M., Influence of acetate on growth kinetics and production control of *Saccharomyces cerevisiae* on glucose and ethanol, Appl. Microbiol. Biotechnol., 24 (1986) 193-198.

2. Rajab.A., Modélisation et conduite automatique de la fermentation de *Saccharomyces cerevisiae*, PhD Thesis, INPL, Nancy, France (1986).

3. Garrido-Sanchez, L.E., Dantigny P. and Pons, M.N., Determination of mineral ion consumptions by ionic HPLC, Biotechnology Techniques, 2 (1998) 17-22.

4. Pons, M.N., Greffe, J.L. and Bordet, J., Fast pH calculations in aqueous solution chemistry, Talanta, 30 (1983) 205-208.

FEEDBACK CONTROL OF MICROBIAL GROWTH PROCESSES WITH NON-MONOTONIC GROWTH KINETICS IN FED-BATCH BIOREACTORS

I.Y. Smets[1], G. Bastin[2] and J.F. Van Impe[1,*]

[1]BioTeC - Department of Food and Microbial Technology, Katholieke Universiteit Leuven
Kardinaal Mercierlaan 92, B-3001 Leuven (Heverlee), Belgium
*Corresponding author Fax: +32-16-32.19.60 E-mail: jan.vanimpe@agr.kuleuven.ac.be
[2]CESAME - Université Catholique de Louvain B-1348 Louvain-la-Neuve, Belgium

Abstract. The aim of this study is to design a *feedback* control algorithm for the limiting substrate flow rate u which forces the substrate concentration to a prespecified value C_S^* in a fed-batch microbial growth process when non-monotonic kinetics apply. This type of control is applicable to a lot of industrial fermentation processes, with the baker's yeast fermentation as a well known example. The specific growth rate μ is assumed to be function of the substrate concentration only. A first approach exploits the availability of on-line measurements of both the substrate and biomass concentrations. A second approach is based on on-line measurements of the biomass concentration only. Noise on the on-line measurements is taken into account.

Introduction

Due to inhibition or repression effects, a lot of microbial growth/production processes exhibit *non-monotonic* kinetics (e.g., *Haldane* kinetics (1), Figure (1)). The feedback control algorithm derived in this paper *(i)* overcomes the ambiguity induced by the non-unique relationship between the specific growth rate μ and the substrate concentration C_S, and *(ii)* forces the substrate concentration C_S to a desired setpoint C_S^* starting from an arbitrary (initial) substrate concentration, even if only noisy (but on-line) measurements of the biomass concentration are available. Different goals of the production process are attained by selecting small or large setpoints for the substrate concentration. For example, in the case of a baker's yeast fermentation a small setpoint favors biomass production while a large setpoint favors ethanol production. In this paper we illustrate the controller design for a small setpoint ($C_S^* < C_{S,\mu}$).

$$\mu = \mu_m \frac{C_S}{C_S + K_P + \frac{C_S^2}{K_I}} \qquad (1)$$

In Expression (1) μ_m [1/h], K_I [g/L] and K_P [g/L] are kinetic constants. In Figure (1) C_S^* and $C_{S,a}^*$ represent the desired setpoint and its associated (undesired) value. $C_{S,\mu}$ is the substrate concentration which maximizes μ.

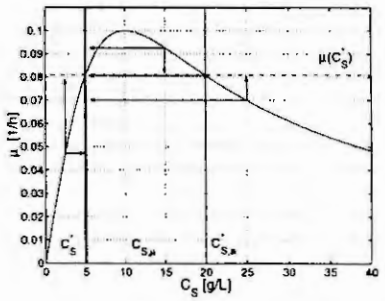


Figure 1: *Haldane* kinetics.

On-line measurements of C_S and C_X

■ Optimal control

For a biomass production process in a fed-batch bioreactor following mass balance equations apply.

$$\begin{aligned} \frac{dC_S}{dt} &= -(\frac{\mu}{Y_{X/S}} + m)C_X + \frac{u}{V}(C_{S,in} - C_S) \\ \frac{dC_X}{dt} &= \mu C_X - \frac{u}{V}C_X \\ \frac{dV}{dt} &= u \end{aligned} \qquad (2)$$

with C_X and C_S [g/L] the biomass and substrate concentration in the reactor respectively, $C_{S,in}$ [g/L] the (fixed) substrate concentration in the influent, V [L] the volume of the reactor, u [L/h] the flow rate, μ [1/h] the specific growth rate (Equation (1)), m [1/h] the specific maintenance coefficient and $Y_{X/S}$ [-] the yield coefficient of biomass on substrate.

The optimal feed rate u^* (in the sense of the minimum principle of Pontryagin) which maximizes the final amount of biomass, is the following [3].

$$u^* = \frac{(\frac{\mu}{Y_{X/S}} + m)C_X}{C_{S,in} - C_S}V \tag{3}$$

This control law keeps the substrate concentration constant. The optimal choice of the substrate setpoint is $C_{S,\mu}$ which maximizes the specific growth rate μ.

■ **Feedback linearizing control: on-line measurements of C_S and C_X**
If substrate concentration and biomass concentration measurements are on-line available, the linearizing control law (4) can be interpreted as the feedforward optimal control (3) plus feedback action [3].

$$u = \frac{(\frac{\mu}{Y_{X/S}} + m)C_X - \lambda(C_S - C_S^*)}{C_{S,in} - C_S}V \tag{4}$$

λ is a strictly positive tuning factor.
The closed loop dynamics of the substrate concentration C_S illustrate the convergence to the desired setpoint C_S^*:

$$\frac{d(C_S - C_S^*)}{dt} = -\lambda(C_S - C_S^*)$$

■ **Simulations**
All simulations are performed using a *continuous time process model* and a *discrete time control* action (ΔT = 1 min). Between two samples the controller is kept constant. In addition, the on-line measurements of C_S and C_X are assumed to be corrupted by (zero mean white) noise. The standard deviation is set equal to $\mathtt{std}(C_X)$ = 0.25 g/L and $\mathtt{std}(C_S)$ = 0.5 g/L. The other simulation parameters are: λ = 10; $C_{S,in}$ = 500 g/L; $Y_{X/S}$= 0.5; m = 0.29 1/h; K_I = 1 g/L ; K_P = 100 g/L; μ_m = 2.1 1/h; $C_{S,\mu} \equiv \sqrt{K_I K_P}$ = 10 g/L; u_{max} = 1 L/h. The setpoint is selected as C_S^* = 5 g/L. As a result, its associated value $C_{S,a}^*$ is equal to 20 g/L. The ability to reach the desired setpoint from different initial substrate concentrations $C_S(0)$ is illustrated in Figure (2).

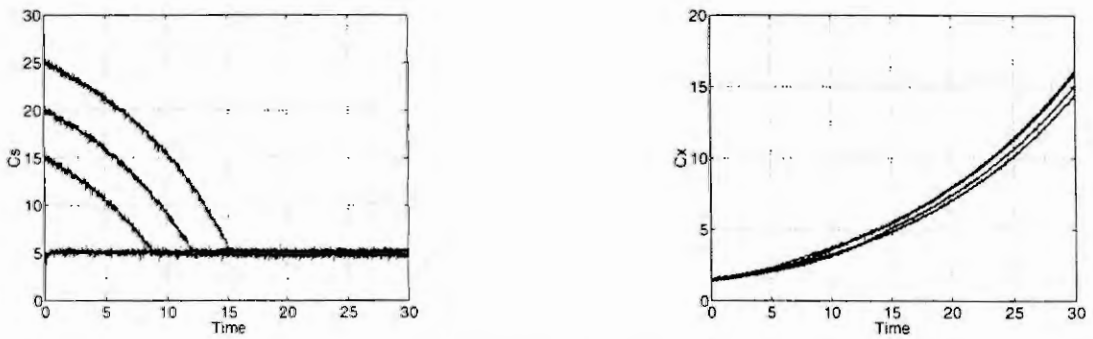


Figure 2: Performance of C_S setpoint controller (4) for various initial substrate concentrations with C_S^* = 5 g/L. Left plot: substrate concentration profiles with respect to time. Right plot: biomass concentration profiles with respect to time.

On-line measurements of C_X

In this section we assume that only on-line measurements of the biomass concentration C_X are available. On-line estimates of the specific growth rate μ can then be supplied by a C_X-based μ-observer (see

[1]). In our research group, on-line measurements of the biomass concentration are provided by the Biomass Monitor (BM 214-M, Aber Instruments LTD, Aberystwyth, UK, [2]). This in-situ monitor relates the capacitance of the medium with the viable biomass concentration. The control objective is reformulated from reaching a desired setpoint for the substrate concentration C_S^* to reaching a desired setpoint for the specific growth rate μ^*. Because of the non-monotonic behavior of the specific growth rate μ as function of substrate concentration C_S, convergence to the associated value $C_{S,a}^*$ (instead of to the desired setpoint C_S^*) is now possible.

$$u = \frac{(\frac{\mu}{Y_{X/S}} + m)C_X - \lambda(\mu - \mu^*)}{C_{S,in} - C_S} V \tag{5}$$

If *Haldane* kinetics (1) are assumed the closed loop dynamics of C_S become ($\lambda > 0$)

$$\begin{aligned} \frac{d}{dt}(C_S - C_S^*) &= -\lambda(\mu - \mu^*) \\ &= -\lambda_{eq}(K_P K_I - C_S C_S^*)(C_S - C_S^*) \quad \text{with} \quad \lambda_{eq} > 0. \end{aligned}$$

Clearly, convergence of the closed loop is determined by the sign of $(K_P K_I - C_S C_S^*)$. In order to compensate for the possibly negative sign of this factor and for convergence to the wrong (associated) value $C_{S,a}^*$ a final adaptation to control law (5) is made by introducing a factor f which equals +1 or -1. The improved algorithm can be summarized as follows.

1. Calculation of factor f.

   If the setpoint C_S^* is smaller than or equal to $C_{S,\mu}$, factor f is equal to +1. If however the specific growth rate is smaller than the desired setpoint μ^* *and* decreasing, factor f switches to -1.

   If the setpoint C_S^* is larger than $C_{S,\mu}$, factor f is equal to -1. If however the specific growth rate is smaller than the desired setpoint μ^* *and* decreasing, factor f switches to +1.

2. Calculation of the flow rate u, based on estimated ($\hat{\mu}$(t)) and on-line measured ($C_{X,m}$(t)) values

$$u = \frac{(\frac{\hat{\mu}}{Y_{X/S}} + m)C_{X,m} - \lambda f(\hat{\mu} - \mu^*)}{C_{S,in} - C_S^*} V \tag{6}$$

   which is bounded by u_{min} (= 0) and u_{max}.

Simulations

Again the control law is implemented in discrete time (ΔT = 1 min, which is a realistic value for the abovementioned Biomass Monitor) with zero mean white noise (std(C_X) = 0.25) on the biomass measurements. The performance of the controller is illustrated in Figure (3). For an initial substrate concentration $C_S(0)$ equal to 25 g/L (dashed line) the factor f is initialized at a value of +1 since the setpoint C_S^* (= 5 g/L) is smaller than $C_{S,\mu}$ (= 10 g/L). The factor f switches to -1 because the estimated specific growth rate $\hat{\mu}$ is smaller than the setpoint μ^* *and* decreasing. f switches back to +1 when $\hat{\mu}$ enters the *attracting* region of μ^*. For an initial substrate concentration $C_S(0)$ equal to 2.5 g/L (full line) the factor f (initialized at a value of +1) does not (have to) switch because $\hat{\mu}$(t) always lies in the region of attraction of μ^*.

Conclusions

In this paper a feedback control algorithm is derived which forces the suhstrate concentration to a prespecified setpoint in fed-batch microbial growth processes with non-monotonic growth kinetics.

If (noisy) measurements of both the substrate concentration C_S and the biomass concentration C_X are on-line available control law (4) (which is a feedback implementation of the optimal control law) has

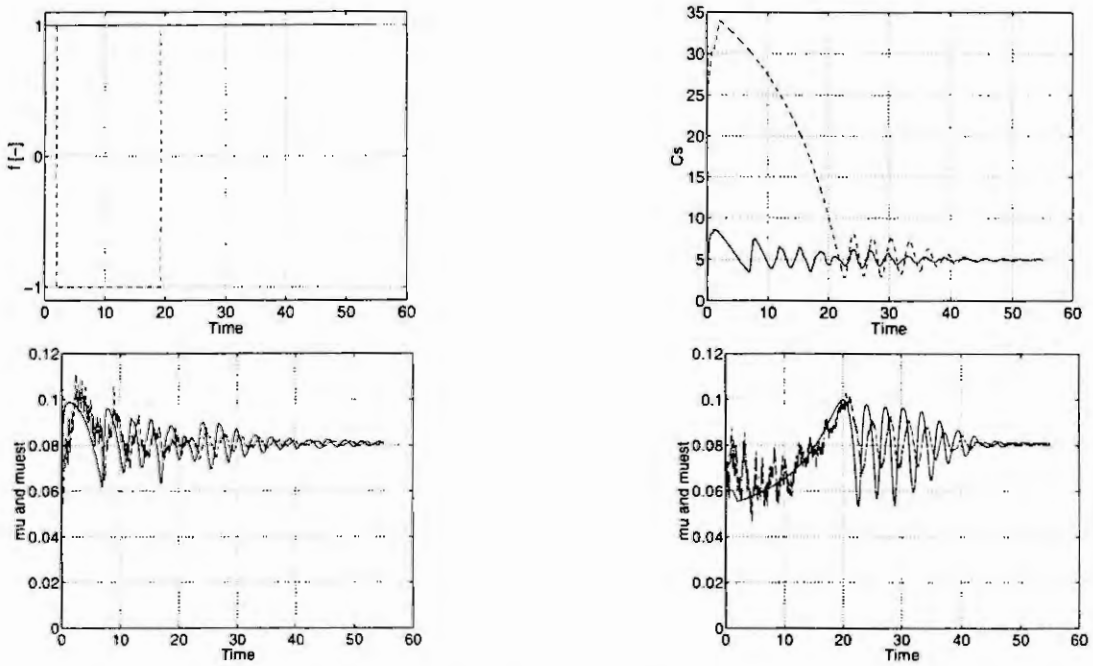


Figure 3: Performance of μ setpoint controller (6) for $\mu^* = \mu(C_S^*) = 0.08$ 1/h with $C_S^* = 5$ g/L for $C_S(0) = 2.5$ g/L (full line) and $C_S(0) = 25$ g/L (dashed line). Upper left plot: switch factor f. Upper right plot: substrate concentration C_S with respect to time. Lower left plot: specific growth rate μ (full line) with respect to time and its estimate $\hat{\mu}$ (dashed line) for $C_S(0) = 2.5$ g/L. Lower right plot: μ (full line) and $\hat{\mu}$ (dashed line) for $C_S(0) = 25$ g/L.

excellent performance. With both state variables on-line available, convergence to the desired substrate concentration setpoint is guaranteed in spite of the non-monotonic behavior of the specific growth rate.

In the (more realistic) case that only (noisy) measurements of the biomass concentration C_X are available, the non-monotonicity of the specific growth rate causes additional difficulties. These are circumvented as follows. The specific growth rate is estimated on-line using the (on-line) biomass measurements. The control objective is reformulated from convergence to a desired substrate concentration C_S^* to convergence to the corresponding specific growth rate $\mu^* = \mu(C_S^*)$. To prevent convergence to the so-called associated substrate concentration $C_{S,a}^*$, a switch factor f is introduced. By doing so, the control algorithm always forces the bioreactor state to the region of attraction around the desired substrate concentration C_S^*.

Acknowledgments

Author Ilse Smets is a research assistant with the Fund for Scientific Research Flanders (FWO). Work supported in part by Projects OT/95/20 and OT/99/24 of the Research Council of the Katholieke Universiteit Leuven and the Belgian Program on Interuniversity Poles of Attraction, initiated by the Belgian State, Prime Minister's Office for Science, Technology and Culture. The scientific responsibility is assumed by its authors.

References

[1] G. Bastin and D. Dochain 1990. *On-line Estimation and Adaptive Controle of Bioreactors.* Elsevier Science Publishing Co., Amsterdam

[2] C.L. Davey 1993. The biomass monitor source book. A detailed user guide. *Aber Instruments Ltd.*, Science Park, Abersystwyth, Wales (UK)

[3] J.F. Van Impe and G. Bastin 1995. Optimal adaptive control of fed-batch fermentation processes. *Control Engineering Practice*, 3(7), 939-954

[4] J.F. Van Impe 1998. Optimal control of fed-batch fermentation processes, In: J. Van Impe, P. Vanrolleghem and D. Iserentant (Eds) *Advanced Instrumentation, Data Interpretation and Control of Biotechnical Processes*, Kluwer Academic Publishers, 319-346

DESIGN OF ROBUST H_∞ ESTIMATOR FOR BIOPROCESSES : APPLICATION TO A FLUIDIZED BED BIOREACTOR

C. Verdier, J-F. Béteau
LAG - Laboratoire d'Automatique de Grenoble
ENSIEG, BP 46, 38402 Saint Martin d'Hères, France
Tel. +33 (0)4 76 82 64 76

Abstract.
An original method is proposed to estimate the state of bioprocesses, by taking into account uncertainties and noises on the system. Hence, a robust H_∞ estimator is presented, where a performance criterion associated to a robust one, leads to solve a min-max problem. This estimator supposes to get a linear time variant model. Therefore, we also propose to transform the general non linear model of bioprocesses into a time varying linear one by using a physical property of the produced biomass rate, obtained from a simply input/output CSTR model. The method is applied to the model of a fluidized bed reactor using anaerobic digestion, and simulation results are discussed.

Introduction

This paper is aimed at proposing a robust H_∞ estimator for a general dynamical nonlinear model of bioprocesses ([1], [6]):

$$\dot{\xi}(t) = K.\rho(\xi,t) + L.\xi(t) + F(t) \qquad (1)$$

where:

- $\xi = [\xi_1 \ldots \xi_n]$ is the state,
- $K.\rho(\xi,t)$ describes the kinetics of biochemical and microbiological reaction which are involved in the reactor. K is the stoechiometric matrix and $\rho(\xi,t)$ represents the biological reaction rates. We note $\mu X(t)$ a produced biomass rate.
- $L.\xi(t)$ is a complex hydrodynamical model which describes the transport dynamics of the components through the reactor. L is a full matrix.
- $F(t)$ is the vector of liquid and gaseous inputs/outputs.

We suppose that the model describes at least the biomass(es) evolution and the substrate(s) degradation. We need to model the product(s) generation in the case where biomass(es) specific growth rates are not proportional.

Our motivation is to take into account uncertainties that we classify into 2 parts:

1. Uncertainties on the biological dynamics, due to the lack of knowledge on biological reaction rates and due to some modeling simplification.
2. Noise measurement and unmeasured disturbances.

A first approach consists in applying a linear transformation on the model (1) in order to obtain a linear model which does not depend on biological dynamics ([1], [6]). This method provides good performances but it supposes that no noise measurements exist and that the hydrodynamical model is exact.
Another approach is to design an Extended Kalman Filter (EKF) for the linearized model of (1). However, it is quite difficult to tune an EKF for a bioprocess because of the lack of knowledge concerning noises on the system. Furthermore, the linearized model is valid only near an equilibrium point and the EKF may diverge if the system is too far from this point. Last but not least, EKF may give biased estimates or even diverge if the initialization of the observer is bad [2].
The approach we propose is to design an optimal estimator for bioprocesses by taking into accounts uncertainties described above. We suppose no a priori structure for the biological reaction rate, like in [1].

I. Modeling

Robust H_∞ estimators only exist for a class of non linear systems, like in [4], where one just consider multiplicative uncertainties on the non linear function. Therefore, we propose to transform the non linear model (1) in order to obtain a linear time varying model, where solutions to the problem of robust H_∞ filtering can be

found [3]. Estimation of the specific growth rate without any knowledge of the bioprocess state is a complicated problem. On the other hand, it appears quite easy to estimate the produced biomass rate $\mu X(t)$, when exists a measurement which is asymptotically proportional to the produced biomass rate.
Let us consider a typical example to illustrate how we can transform $\mu X(t)$ into a time varying state without any assumption on the structure of the specific growth rate μ. Indeed, in the case of a substrate transformation, the dynamics can be written as following, by considering a CSTR model:

$$\frac{dS(t)}{dt} = -\frac{\mu X(t)}{y_1} + D.\left(S_{in}(t) - S(t)\right) \tag{2}$$

where y_1 is the yield coefficient substrate/biomass, D is the dilution rate and S_{in} is the input substrate concentration.
Remark: In the case of a CSTR model, we have: $L = D.I$, with I the identity matrix.

We define, for each operating point, the produced biomass rate by the following equation:

$$\mu X_o(t+T) = y_1.D.\left(S_{in} - S(t)\right) \tag{3}$$

We just consider one sampling period T between the variation on the operating conditions and the response on μX_o, which is justified since μX_o is the asymptotically produced biomass rate.
The transient response μX_t of the produced biomass rate can be derived from the equilibrium produced rate μX_o by the following model expressed in Laplace transform:

$$\mu X_t(s) = K.\mu X_o(s) + F(s).\mu X_0(s) \tag{4}$$

The model of μX_t can generally be expressed by the sum of 2 terms: the first one describes the effects of the operating conditions variations; the second one describes the transient response of μX_t between 2 operating points, and is expressed as a function of the new equilibrium point filtered by F(s).
The discrete-time model of the transient response of the produced biomass rate is obtained by performing the z transform of (4). Hence, by combining eq. (3) and (4), we obtain:

$$\mu X_t(k+1) = K.D.y_1.\left(S_{in} - S(k)\right) + F_{den}(z^{-1}).\mu X_t(k) + F_{num}(z^{-1}).\mu X_o(k) \tag{5}$$

Remark: The strategy we propose for the transformation of the model (1) is valid whatever the equilibrium point. For example, the washout point is reached with the condition: $S_{in} = S(k)$.

Uncertainties on the biological dynamics lead to variations on K and on the parameter(s) of the filter F. Noise measurement is derived from the knowledge of sensor uncertainties and an a priori uncertainty on the modeling error is fixed. We also define uncertainties on the input flow rate, derived from the knowledge of the sensor accuracy.
Thus, we are able to write the uncertain model of a bioprocess by increasing the number of states with the dynamics of the produced biomass rate (eq. 3 and 5). Finally, we obtain a linear time varying (LTV) model:

$$\xi(k+1) = A_k.\xi(k) + B_k.U(k) + \Delta\varepsilon_k \tag{6}$$

where $\Delta\varepsilon_k$ is the term describing uncertainties.

We have developed in our research team a non linear model for a fluidized bed bioreactor using anaerobic digestion ([5]). This model has been validated on a experimental pilot and is expressed by the following system:

$$\frac{dX(t)}{dt} = (\mu(t) - k_d)X(t) \text{ and } \begin{cases} \dfrac{dA_i(t)}{dt} = -\dfrac{\mu(t).X(t)}{y_1} + \left[l_{ij}\right].\left[A_i(t)\right]^T + l_{0i}.A_{inr} \\ \dfrac{dZ_i(t)}{dt} = \left[l_{ij}\right].\left[Z_i(t)\right]^T + l_{0i}Z_{inr} \\ \dfrac{dIC_i(t)}{dt} = -\dfrac{\mu(t).X(t)}{y_1 y_2}\left[\dfrac{K_H - 2CO_{2d}}{K_H - CO_{2d}}\right] + \left[l_{ij}\right].\left[IC_i(t)\right]^T + l_{0i}.IC_{inr} \end{cases} \tag{7}$$

with $\xi = \left[X, A_i, Z_i, IC_i\right]$. Signification of all parameters can be found in [5] or [6].
We have applied the transformation we propose to this model, where the specific growth rate is described by an Haldane law. The filter F is modeled by a first order depending on the design parameter a:

$$\begin{aligned} \mu X_o(k+1) &= y_1.D.\left(S_{in} - S(k)\right) \\ \mu X_t(k+1) &= K.D.y_1.\left(S_{in} - S(k)\right) + (1 - aT).\mu X_t(k) + (aT - K).\mu X_o(k) \end{aligned} \tag{8}$$

The first term $K.\mu X_o(k+1)$ represents the produced biomass rate which depends on the physico-chemical equilibrium, considering as instantaneous. The second term describes the production rate dynamics as a function

of $\mu X_a(k)$. The parameters a and K are uncertain, depending of the magnitude of perturbations: a variation on a means that the model used to described μX_t is not exact, while a variation on K means an uncertainty on the physico-chemical equilibrium (on pH, for example).

Thus, the non linear model (7) can be represented by the (LTV) one, with $\xi = \left[X, \mu X_0, \mu X_t, A_i, Z_i, IC_i \right]$

$$A_k = \begin{bmatrix} 1-T.k_d & 0 & T & 0_{1x4} & 0_{1x4} & 0_{1x4} \\ 0 & 0 & 0 & 0_{1x3} \; -D.y_1 & 0_{1x4} & 0_{1x4} \\ 0 & aT-K & 1-aT & 0_{1x3} \; -K.D.y_1 & 0_{1x4} & 0_{1x4} \\ 0 & 0 & -\frac{T}{y_1}\begin{bmatrix}1\\1\\1\\1\end{bmatrix} & \psi & 0_{4x4} & 0_{4x4} \\ 0 & 0 & 0_{4x1} & 0_{4x4} & \psi & 0_{4x4} \\ 0 & 0 & \frac{T}{y_1.y_2}.\frac{Q_{CH_4}-Q_{CO_2}}{Q_{CH_4}} & 0_{4x4} & 0_{4x4} & \psi \end{bmatrix} \quad B_k = \begin{bmatrix} 0 & 0 & 0 \\ D.y_1 & 0 & 0 \\ K.D.y_1 & 0 & 0 \\ T.L_0.\frac{Q_{rec}}{Q} & 0_{4x1} & 0_{4x1} \\ 0_{4x1} & T.L_0.\frac{Q_{rec}}{Q} & 0_{4x1} \\ 0_{4x1} & 0_{4x1} & T.L_0.\frac{Q_{rec}}{Q} \end{bmatrix} \tag{9}$$

$$\psi = I + T.L_1 + T.\left(\frac{Q_{in}}{Q}\right).\left[0_{4x3} \quad L_0\right]$$

We present simulation results for biomass and produced biomass rate for a substrate step from S_{in} = 0.11 mol.l^{-1} to 0.22 mol.l^{-1}. We have considered the non linear model and the LTV model without uncertainties.

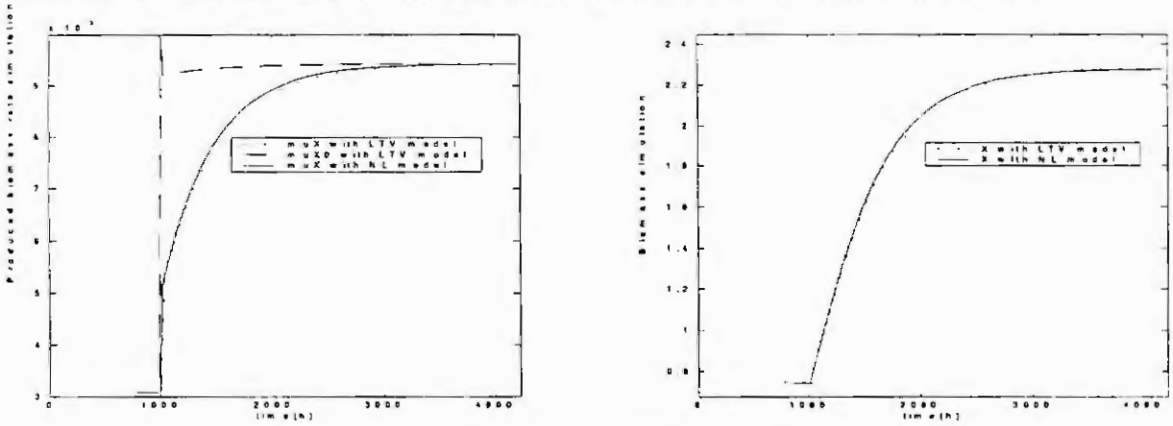


Figure 1: Validation of the LTV model

II. Design of robust H_∞ estimator

The robust estimator problem is to minimize the estimation error e for an entire family of possible plants, defined from the nominal plant P and the uncertainties Δ. The vector r define all the input perturbations and $\xi_0 - \hat{\xi_0}$ represents the uncertainty due to the estimation initialization.

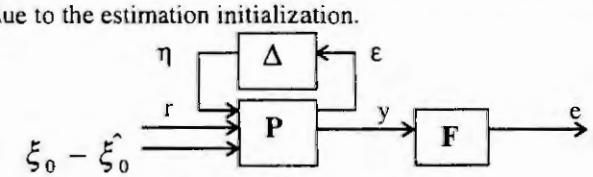


Figure 1: Robust estimation problem

The objective is to respect a performance criterion, provided the model perturbations Δ have bounded l_2 norm:

- performance criterion:
$$\sup_{(r,\xi_0-\hat{\xi_0})\neq 0} J_1 \equiv \frac{\|e\|^2}{\|r\|^2 + \left\|\xi_0 - \hat{\xi_0}\right\|^2} < \gamma^2 \tag{10}$$

- robust criterion:
$$\|\Delta\|^2 \equiv \sup_{\varepsilon \neq 0} \frac{\|\eta\|^2}{\|\varepsilon\|^2} < \gamma^{-2} \tag{11}$$

We incorporate the model perturbations into the performance criterion and we obtain ([3]):

$$\sup_{(r,\eta,\xi_0-\hat{\xi_0})\neq 0} J_2 \equiv \frac{\|e\|^2 + \|\varepsilon\|^2}{\|r\|^2 + \|\eta\|^2 + \|\xi_0\|^2_{\bar{X}_0} + \left\|\xi_0 - \hat{\xi_0}\right\|^2} < \gamma^2 \tag{12}$$

The solution to this minimization leads to define a min-max problem, where we seek to minimize an objective function with respect to the state estimate in the presence of the worst possible inputs and initial state:

$$\min_{\hat{\xi}} \max_{r,\eta,\xi_0} J_3 = \|e\|^2 + \|\varepsilon\|^2 - \gamma^2\left(\|r\|^2 + \|e\|^2 + \|\xi_0\|^2_{X_0} + \|\xi_0 - \hat{\xi}_0\|^2_{P_0^{-1}}\right) \quad (10)$$

One possible approach to solve this kind of problem comes from [3] in terms of 2 Riccati equations.

This robust H_∞ estimator is developed on the uncertain model of the fluidized bed bioreactor. The frequency response of the estimator depends on the design parameter γ. Decreasing this parameter trades off nominal performances in term of transfer function from noise to estimation error to provide robustness to disturbance and plant modeling error. Indeed, this estimator is an extension of estimator and Kalman filter. In the case where $\gamma \to \infty$, one recovers the Kalman filter.

For our model, an observability study shows that the biomass is unobservable with the proposed model if we don't measure it. Nevertheless, the biomass dynamics can be reconstruct in a second stage with all the others states. The maximum singular value of the transfer function G from the perturbation input to estimation error for the nominal model and the perturbed one is represented below:

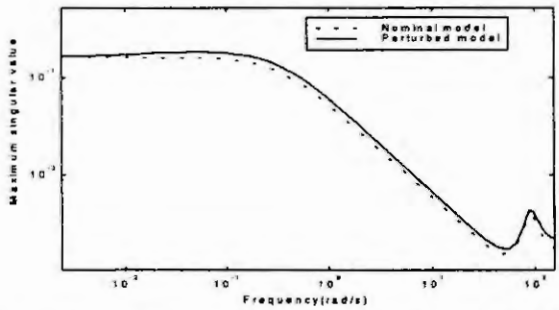


Figure 2 : Frequency response of the transfer G for the nominal model and the perturbed one.

We have considered in this case an error of 10% on the design parameters *a* and K, and a modeling error of 15% on the methane/carbon dioxide ratio. The comparison between the frequency responses of the estimator for the nominal model and the perturbed one shows the robustness of the algorithm to such structural uncertainties and modeling approximations, in particularly for the higher frequency noise.

Conclusion

This paper proposes a strategy for the design of a robust H_∞ estimator dedicated to bioprocesses. This strategy requires to transform the general non linear model of bioprocesses into a linear time variant one, from a measurement which is asymptotically proportional to the produced biomass rate. The estimator depends on the design parameter γ, which quantifies the tradeoff between the robustness and the performance of the estimator. The method is applied to the model of a fluidized bed bioreactor using anaerobic digestion.

It would be now interesting to take into account more uncertainties, in order to evaluate the influence of each uncertainty on the robustness. Furthermore, the proposed method would be applied on other bioprocesses.

References

[1] Bastin G. and Dochain D. - On-line Estimation and Adaptive Control of Bioreactors: I, II, III - Elsevier Publisher, 1990.

[2] Ljung L. - Asymptotic behavior of the extended Kalman filter as a parameter estimator for linear systems . In IEEE Trans. On Automatic Control, AC-24(1): 36-50, 1979

[3] Rami, S. Mangoubi, Brent, D. Appleby and George, C. Verghese - Stochastic Interpretation of H_∞ and Robust Estimation. In: 33rd Conference on Decision and Control, 1994, Lake Buena Vista,, 3943-3948.

[4] De Souza, Carlos E., Xie, Lihua and Wang, Youyi - H_∞ filtering for a class of uncertain nonlinear systems. In systems & Control Letters 20, 1993, 419-426.

[5] Otton, V. - Modélisation et Analyse d'un Procédé à Paramètres Répartis. PhD report, INPG/LAG, 1997

[6] Verdier C. and Béteau J-F. - Asymptotic Observer for a Distributed Parameters Model: Application to a fluidized Bed Reactor - In: European Control Conference, 1999.

SYSTEMATIC MODELLING METHODOLOGY FOR SIMULATION AND STATE ESTIMATION OF BIOPROCESSES

Ph. Bogaerts[1], A. Vande Wouwer[2] and R. Hanus[1]
[1]Control Engineering and System Analysis Department, Université Libre de Bruxelles
Ave. F.-D. Roosevelt, 50 C.P. 165/55 B-1050 Brussels (Belgium)
Tel. 32-2-650.26.75 Fax. 32-2-650.26.77 E-mail: pbogaert@labauto.ulb.ac.be
[2]Control Engineering Department, Faculté Polytechnique de Mons
Bld. Dolez, 31 B-7000 Mons (Belgium)
Tel. 32-65-37.41.41 Fax. 32-65-37.41.36 E-mail: vdw@autom.fpms.ac.be

Abstract. This contribution focuses on a systematic methodology for mathematical modelling of bioprocesses with a view to their simulation and/or their state estimation. The main steps of the methodology are reviewed (macroscopic reaction network, mass balances, three-step procedure for the parameter identification) and a novel concept is sketched, namely the parameter estimation of a model to be used for state estimation.

1 Introduction

Biotechnology and bioprocesses are of great interest as they lead to several valuable products, such as vaccines, antibodies, food products, etc. In order to optimize productivity and quality, it is useful to derive simulation models, which could also be used for on-line optimal control [2,9]. Another reason for deriving a mathematical model is the design of state observers (or software sensors) allowing some unmeasured states to be reconstructed on-line [2]. This is especially interesting in the field of bioprocesses where the hardware measurements often present several drawbacks (price, destruction of the samples, time delay, discrete- (instead of continuous-) time results, sterilization, etc.).

In both cases (simulation and state estimation), it is necessary to select an appropriate model structure and to identify its parameters on the basis of some measurements. Numerous solutions have already been proposed in the literature, using different levels of complexity [1,2]. However, these procedures often present several drawbacks with regard to the kinetic model structures and to parameter identification. To overcome these usual problems, a systematic methodology is proposed in this paper (for details, see also [3]). The selection of a model structure is considered in Section 2 and parameter identification is dealt with in Section 3. The model obtained with this methodology can directly be used for simulation, but can also be the basis of a state observer synthesis. However, if the model part relative to the measured states is not sufficiently sensitive w.r.t. the measured one, the state observer might lead to accurate estimates of the measured states and to very poor estimates of the non measured states. A new combined cost function for the parameter identification is proposed in Section 4, taking into account this problem of state estimation sensitivity.

2. Model structure

As the modelling aims concern the simulation, process optimization, state estimation (and control) of bioprocesses, it is necessary to limit the complexity of the model structure. A basic tool of the methodology is the concept of reaction network, describing the main macroscopic phenomena occurring in the culture [2]. Note that this kind of network does not need to respect the elementary mass balances.

$$\sum_{i \in R_k} \left(-\nu_{i,k}\right)\xi_i \rightarrow^{\varphi_k} \sum_{j \in P_k} \nu_{j,k}\xi_j \qquad k \in [1,M] \tag{1}$$

where M is the number of reactions, ξ_i the i-th component, φ_k the reaction rate, $\nu_{i,k}$ and $\nu_{j,k}$ the pseudo-stoichiometric (or yield) coefficients (positive when associated to a component which is produced, negative when it is consumed).

The model structure consists of the mass balances for each of the component ξ_i appearing in (1). This system of mass balances can be written in the following matrix form:

$$\frac{d\xi(t)}{dt} = K\varphi(\xi,t) - D(t)\xi(t) + F(t) - Q(t) \tag{2}$$

where $\xi \in \Re^N$ is the vector of concentrations, $K \in \Re^{N \times M}$ is the pseudo-stoichiometric coefficients matrix, $\varphi \in \Re^M$ is the vector of reaction rates, $D \in \Re$ is the dilution rate, $F \in \Re^N$ is the vector of external feed rates and $Q \in \Re^N$ is the vector of gaseous outflow rates.

Sufficient conditions of bounded input - bounded state stability are proposed in [3,4]. Concerning the reaction rates φ_j, a new general kinetic model structure has been proposed:

$$\varphi_j(\xi_1,\cdots,\xi_N,t) = \alpha_j \prod_{k \in R_j^*} \xi_k^{\gamma_{kj}}(t) \prod_{l \in P_j^*} e^{-\beta_{lj}\xi_l(t)} \qquad j \in [1,M] \tag{3}$$

where $\alpha_j > 0$ is a kinetic constant (function, if necessary, of any physical influence different from the component concentrations, e.g., the temperature dependance according to an Arrhenius law), R_j^* the set of indices of the components which activate the reaction j (reactants, catalysts and auto-catalysts), P_j^* the set of indices of all the components appearing in reaction j (or even, if necessary, in other reactions of the scheme), $\gamma_{kj} > 0$ the activation coefficient of component k in reaction j and $\beta_{lj} \geq 0$ the inhibition coefficient of component l in reaction j.

For a given reaction network (1), and on the basis of measurements of the vector ξ, it is necessary to identify the values of the pseudo-stoichiometric coefficients and of the kinetic parameters (together with the initial concentrations $\xi(0)$ which might be corrupted by noise as well as for any other sampling instant). This problem is tackled in the next section.

3. Parameter identification

A three-step procedure has been proposed in [3]. The basics of this methodology are given hereafter.

First step: estimation of the pseudo-stoichiometric coefficients (independently of the kinetic coefficients)

According to the decoupling method proposed in [6], it is possible to find a full row rank submatrix $K_a \in \Re^{p \times M}$ (where $p = \text{rank}\,K$) of a partition $K^T = \begin{bmatrix} K_a^T & K_b^T \end{bmatrix}$. Hence, there exists a unique solution $C \in \Re^{(N-p) \times p}$ to the matrix equation

$$CK_a + K_b = 0_{N-p,M} \tag{4}$$

It is then possible to define an auxiliary vector

$$z = C\xi_a + \xi_b \tag{5}$$

whose dynamics are independent of the reaction rates $\varphi(\xi)$:

$$\frac{dz(t)}{dt} = -D(t)z(t) + Cu_a(t) + u_b(t) \tag{6}$$

(where $u^T = \begin{bmatrix} u_a^T & u_b^T \end{bmatrix}$ corresponds to the partition $K^T = \begin{bmatrix} K_a^T & K_b^T \end{bmatrix}$). C can be estimated on the basis of relation (5) where $z(t)$ is obtained by integration of (6):

$$z(t) = \left(z(0) + \int_0^t \left(Cu_a(\tau) + u_b(\tau) \right) e^{\int_0^\tau D(\kappa)\,d\kappa}\, d\tau \right) e^{-\int_0^t D(\kappa)\,d\kappa} \tag{7}$$

Considering the particular (but very general case) where $p = \text{rank}\,K = M$, a necessary and sufficient condition in order to be able to univoquely determine K_a and K_b from relation (4) and the knowledge of C, is that there exists a partition $K^T = \begin{bmatrix} K_{a'}^T & K_{b'}^T \end{bmatrix}$ (with $K_{a'} \in \Re^{M \times M}$ invertible) such that $K_{a'}$ does not contain any unknown coefficient of K.

When this condition is fulfilled, several solutions are proposed in [3] in order to estimate the matrix K on the basis of this decoupling method. One of the solutions consists in using a maximum likelihood criterion allowing to take into account all the measurement errors (for each signal and each sample time, including the initial one). Defining a vector ϑ_C containing all the unknown parameters of C, its maximum likelihood estimation is given by

$$\hat{\vartheta}_C = \underset{\vartheta_C}{ArgMin} \frac{1}{2} \sum_{s=1}^{S} \sum_{k=1}^{N_s} \left(Y_{m,s,k} - \vartheta_s^T(\vartheta_C)\varphi_{m,s,k} \right)^T \left(Q_{Y,s,k} + \vartheta_s^T(\vartheta_C) Q_{\varphi,s,k} \vartheta_s(\vartheta_C) \right)^{-1} \left(Y_{m,s,k} - \vartheta_s^T(\vartheta_C)\varphi_{m,s,k} \right) \tag{8}$$

where $Y_{s,k} = \vartheta_s^T(\vartheta_C)\varphi_{s,k}$, $Y_{s,k} = \eta_{b,s}(t_{s,k}) \in \Re^{N-p}$, $\varphi_{s,k}^T = \left[-\eta_{a,s}^T(t_{s,k}) \quad e^{-\int_0^{t_{s,k}} D(\kappa)\,d\kappa} \right] \in \Re^{1 \times (p+1)}$,

$$\theta_s^T(\theta_C) = \begin{bmatrix} C^{(1,:)}(\theta_C) & z_s^{(1)}(0) \\ \vdots & \vdots \\ C^{(N-p,:)}(\theta_C) & z_s^{(N-p)}(0) \end{bmatrix} \in \Re^{(N-p)x(p+1)}$$ ($t_{s,k}$ being the k^{th} sample time of the s^{th} experiment)

and $Y_{m,s,k} = Y_{s,k} + e_{Y,s,k}$, $\varphi_{m,s,k} = \varphi_{s,k} + e_{\varphi,s,k}$, $E[e_{Y,s,k}] = 0$, $E[e_{\varphi,s,k}] = 0$, $E[e_{Y,s,k}e_{Y,t,l}^T] = Q_{Y,s,k}\delta_{s,t}\delta_{k,l}$, $E[e_{\varphi,s,k}e_{\varphi,t,l}^T] = Q_{\varphi,s,k}\delta_{s,t}\delta_{k,l}$, $E[e_{Y,k}e_{\varphi,l}^T] = 0 \quad \forall k,l$ (Gaussian error distributions).
The covariance matrix of the parameter estimation errors can also be estimated (see [3]).

Second step: first estimation of the kinetic coefficients
The kinetic model structure (3) can be linearized w.r.t. its parameters thanks to a logarithmic transformation. This enables to find a linear least squares estimate of the kinetic coefficients (which is independent of any initial guess):

$$\hat{\theta}_{cin}^{(j)} = \underset{\theta_{cin}^{(j)}}{ArgMin} \frac{1}{2}\sum_{s=1}^{S}\sum_{k=1}^{N_s}\left(Y_{m,s,k}^{(j)} - \varphi_{s,k}^{(j)T}\theta_{cin}^{(j)}\right)^2 \qquad j \in [1,M] \tag{9}$$

where $Y_{s,k}^{(j)} = \ln\hat{\varphi}_j(t_{s,k})$, $\varphi_{s,k}^{(j)T} = \left[1 \;\; \ln\xi_{k,1}(t_{s,k}) \;\; -\xi_{l,1}(t_{s,k})\right]$, $\theta_{cin}^{(j)T} = \left[\ln\alpha_j \;\; \gamma_{k,j} \;\; \beta_{l,j}\right]$.
Note that, in the very general case where $p = \text{rank}K = M$, estimates of the reaction rate $\hat{\varphi}_j(t_{s,k})$ can be obtained with the relation

$$\hat{\varphi}(\xi(t_{s,k})) = \hat{K}_a^{-1}\left(\left(\frac{d\xi_a(t_{s,k})}{dt}\right)^{\wedge} + D(t_{s,k})\xi_a(t_{s,k}) - F_a(t_{s,k}) + Q_a(t_{s,k})\right) \tag{10}$$

where the estimate of the derivative can, for instance, be computed by the analytical derivation of an interpolation model for the vector $\xi_a(t)$. The estimates $\hat{\theta}_{cin}^{(j)}$ are based on unreliable assumptions on the measurement errors (errors only on $Y_{s,k}^{(j)}$, with constant standard deviation) and on estimates of the signal derivatives. Therefore, these estimates are just considered as a (unique and systematic) initial guess for the last step of the identification.

Third step: final estimation of the kinetic coefficients (and of some initial concentrations)
Let $\xi(t,\xi_s(0),u(t))$ be the solution of (2) for initial conditions $\xi_s(0)$ (of experiment s) and input trajectories $u^T(t) = \left[D(t) \;\; F_1(t) \;\; \cdots \;\; F_M(t)\right]$.
Denoting θ the vector containing all the kinetic coefficients together with the initial conditions $\xi_{a,1}(0), \ldots, \xi_{a,S}(0)$, this last step provides the maximum likelihood estimates

$$\hat{\theta} = \underset{\theta}{ArgMin} \frac{1}{2}\sum_{s=1}^{S}\sum_{k=1}^{N_s}\left(\xi_m(t_{s,k}) - g(t_{s,k},\xi_s(0),u(t_{s,k}))\right)^T Q_{s,k}^{-1}\left(\xi_m(t_{s,k}) - g(t_{s,k},\xi_s(0),u(t_{s,k}))\right) \tag{11}$$

where $\xi_m(t_{s,k})$ are the available measurement samples (with error covariance matrix $Q_{s,k}$). Note that the other initial conditions $\xi_{b,1}(0), \ldots, \xi_{b,S}(0)$ can be deduced from relation (5). In this last step, the covariance matrix of the parameter estimation errors can also be estimated (see [3]).

4. Identification for state estimation

The model obtained with the three-step procedure described above can directly be used for simulation of the bioprocess. It can also be used for the synthesis of a state observer. Several techniques have already been used in the field of bioprocesses (asymptotic observers [2], extended Kalman filters [2], extended Luenberger observers [2], high gain observers [7], full horizon observers [3,5], etc.). Concerning all the techniques using the complete model and allowing to estimate the whole state of the model (i.e. all the above mentioned ones except the asymptotic observers), a necessary condition is the system to be observable. In the field of nonlinear systems, this concept of observability is depending on the inputs.
Let us consider a nonlinear model of the form

$$\begin{cases} \dfrac{dx(t)}{dt} = f(x(t),u(t)) \qquad x(0) = x_0 & (12) \\ y(t) = Cx(t) & (13) \end{cases}$$

where $x(t) \in \Re^n$ is the state vector, $u(t) \in \Re^l$ is the input vector, $y(t) \in \Re^m$ is the measured output vector, f is a nonlinear function and $C \in \Re^{mxn}$ is the measurement matrix. Let $y(t,x(0),u(t))$ denote the output trajectory

corresponding to the initial condition $x(0)$ and to the input trajectory $u(t)$. The system {(12),(13)} is observable if, for any couple of different initial conditions $x(0)$ and $x'(0)$, there exists an input $u(t)$ and a time $0<t<\infty$ for which the outputs $y(t,x(0),u(t))$ and $y(t,x'(0),u(t))$ are different. Uniformly observable systems have the particularity that each admissible input to the system is universal, i.e. allows any couple of different initial states to be distinguisbed. Locally U-uniform observability in $x(0)$ is restricted to a neighbourhood $V(x(0))$ of $x(0)$ and to inputs remaining in an admissible domain U. Any locally U-uniformly observable multiple input-multiple output system can be written in the following canonical form (see [8] for the single output case):

$$\begin{cases} \dfrac{d}{dt}\left[x_1^T \ \cdots \ x_i^T \ \cdots \ x_q^T\right] = \left[f_1^T(u,x_1,x_2) \ \cdots \ f_i^T(u,x_1,\cdots,x_{i+1}) \ \cdots \ f_q^T(u,x_1,\cdots,x_q)\right] \\ y(t) = x_1 \end{cases} \tag{14}$$

$\forall i \in \{1,\cdots,q\}$, $x_i \in \Re^{n_i}$, $n_1 \geq n_2 \geq \cdots \geq n_q$ and $\sum_{1 \leq i \leq q} n_i = n$, $\forall i \in \{1,\cdots,q-1\}$, $\forall (u,x) \in U \times \Re^n$: $\operatorname{rank} M_i(u,x) = n_{i+1}$

(with $M_i(u,x) = \left(\dfrac{\partial f_i(u,x)}{\partial x_{i+1}}\right)^T \left(\dfrac{\partial f_i(u,x)}{\partial x_{i+1}}\right) \in \Re^{n_{i+1} \times n_{i+1}}$). If some matrices $M_i(u,x)$ are invertible but ill conditioned, the system will be theoretically (i.e. structurally) observable but a difference in the initial states will be hard to detect in the output trajectories. This observation leads to the definition of a state estimation sensitivity function $S(u,x)$ quantifying this ability to detect, in the output trajectories, any differences in the initial states. This "measure of observability" should be a scalar function of the matrices $M_i(u,x)$. Assuming that the state trajectory $x(t)$ is measured at the sample times t_k ($k \in [1,N]$), the chosen state estimation sensitivity function is

$$S(u,x) = \sum_{k=1}^{N} \sum_{i=1}^{q-1} \left(\operatorname{cond}\left(M_i(u(t_k),x(t_k))\right)\right)^{0.5} \tag{15}$$

where "cond" represents the condition number of the matrix, i.e. the ratio of its largest to its smallest singular values ($\forall M$, $1 \leq \operatorname{cond} M < +\infty$). Based on this concept, a new form of the cost function to be minimized in the parameter identification procedure can be defined. This cost function consists of a weighted sum of a conventional maximum likelihood criterion (F_{ml}) with the observabilty measure ($F_{obsv} = S(u,x)$ given in (15)):

$$\hat{\theta} = \underset{\theta}{Argmin} F(\theta) = \underset{\theta}{Argmin}\left\{F_{ml}(\theta) + \lambda F_{obsv}(\theta)\right\} \tag{16}$$

In the case of the methodology proposed in Section 3, the cost function of the third step (11) should be extended by the term $\lambda F_{obsv}(\theta)$ of relation (16), in order to identify a model with a view to state estimation.

4. Conclusion

The main steps of a general and systematic methodology for modelling bioprocesses have been reviewed. A new cost function has been proposed for parameter identification with a view to state estimation. Promising results have already been obtained with this latter result (applied on animal cell cultures) and will soon be published.

References

1. Bailey, J. E., Mathematical modeling and analysis in biochemical engineering: past accomplishments and future opportunities. Biotechnol. Prog., 14 (1998), 8-20.
2. Bastin, G. and Dochain D., On-line estimation and adaptive control of bioreactors. Elsevier, Amsterdam, 1990.
3. Bogaerts, Ph., Contribution à la modélisation mathématique pour la simulation et l'observation d'états des bioprocédés. PhD thesis, Université Libre de Bruxelles, Brussels, 1999.
4. Bogaerts, Ph., Castillo, J. and Hanus, R., A general mathematical modelling technique for bioprocesses in engineering applications, System Analysis Modeling Simulation, 35 (1999), 87-113.
5. Bogaerts, Ph. and Hanus, R., On-line estimation of biomass concentration in CHO animal cell cultures, Proc. of the Third International Symposium on Mathematical Modelling and Simulation in Agricultural and Bio-Industries ($M^2SABI'99$), Uppsala (Sweden), 1999, 97-102.
6. Chen, L. and Bastin, G., Structural identifiability of the yield coefficients in bioprocess models when the reaction rates are unknown. Math. Biosci., 132 (1996), 35-67.
7. Gauthier, J. P., Hammouri, H., and Othman, S., A simple observer for nonlinear systems - Application to bioreactors. IEEE Trans. Automat. Contr., 37 (1992), 875-880.
8. Gauthier, J.-P. and Kupka, I., Observability and observers for nonlinear systems. Siam Journal Control and Optimization, 32 (4), (1994), 975-994.
9. Van Impe, J. and Bastin G., Optimal adaptive control of fed-batch fermentation processes. Contr. Eng. Practice, 3 (No 7) (1995), 939-954.

Improved Theoretical Identifiability of Model Parameters by Combined Respirometric – Titrimetric Measurements. A Generalisation of Results

Britta Petersen[1,2], Krist Gernaey[1] and Peter Vanrolleghem[1,#]

[1] BIOMATH Department, University of Gent, Coupure Links 653, B-9000 Gent (Belgium)
[2] EPAS NV, Technologiepark 3, B-9052 Gent-Zwijnaarde (Belgium)
[#] Corresponding author : Tel. +32 9 264 59 32, Fax +32 9 223 49 41, E-mail: Peter.Vanrolleghem@rug.ac.be

Abstract. The theoretical identifiability of the two-step nitrification model was studied with respirometric and titrimetric outputs. Most remarkable result is that the autotrophic yield becomes uniquely identifiable when combined data are considered. Furthermore, it is illustrated how the identifiability results can be generalised by applying a set of ASM1 matrix based generalisation rules. It appears that the identifiable parameter combinations can be predicted based on knowledge of the process model under study (in ASM1-like matrix representation), the measured variables and the biodegradable substrate considered.

Introduction

Monod-type growth kinetics are most often applied to describe wastewater degradation processes [6]. The evaluation of the theoretical identifiability of model parameters prior to practical model application, e.g. in the frame of parameter estimation or model calibration, is very important, and is based on the model structure and on the measured variables [3]. Perfect noise-free data are assumed in a theoretical identifiability study whereas in practice the data may be noise corrupted. As a result, parameters may be unidentifiable in practice, even if they are theoretically identifiable [7].

In a study on heterotrophic substrate degradation via the Monod model [7], measurements of both substrate and biomass were assumed to be available, and in that case all parameters were proven to be theoretically identifiable. In another study growth was neglected in the model, and oxygen uptake rate data were considered [3]. In the latter situation only certain parameter combinations were theoretically identifiable. Furthermore, the theoretical identifiability of Monod kinetics for a denitrification model was analysed assuming steady state with respect to growth [1]. These results also confirmed that depending on the measured state variables (nitrate, nitrite, carbon substrate) different parameter combinations were identifiable.

The theoretical identifiability analysis of the two-step nitrification process (Monod kinetics) considering outputs from respirometry (oxygen uptake rates, r_O) and titrimetry (cumulative proton production, Hp) is the objective of this study. Model structures with and without biomass growth are considered. The theoretical identifiability was studied by nsing the Taylor series expansion method, and symbolic manipulations were carried out with MAPLE V. Finally, the results of the identifiability study are generalised based on the stoichiometric matrix of the Activated Sludge Model No.1 [6].

Theoretical background

The basic assumptions of the Taylor series expansion approach are that the output vector and its derivatives with respect to time at the initial time t=0 are assumed to be known and unique. The identifiability analysis is thus reduced to a determination of the solutions for the parameters in a set of (non-linear) algebraic equations. A sufficient condition for the model to be theoretically identifiable is that a unique solution exists for the parameters [9,11].

The model under study (Table 1) is based on ASM1 [6], with some modifications [5]. Nitrification takes place in two steps: (1) oxidation of ammonium (S_{NH}) to nitrite (S_{NO2}) and (2) oxidation of nitrite to nitrate (S_{NO3}). Both nitrification steps can be monitored by oxygen uptake measurements whereas it is only during the first step that protons are produced. Thus, only the first step can be characterised by proton production measurements. Measurements of oxygen uptake rate (r_O) can be carried out via respirometry [2,10], and proton production (Hp) can be quantified via a titrimetric technique [4].

Figure 1 illustrates a typical data set obtained following the addition of S_{NH} to a batch reactor filled with activated sludge at t=0. Concentration profiles were simulated using the model of Table 1. The S_{NH} concentration decreases while S_{NO2} builds up because, for this example, the second nitrification step is slower than the first step. The oxidation of S_{NH} results in a certain oxygen uptake rate (defined as exogenous respiration rate, $r_{O,ex}$). The build-up of S_{NO2} causes the tail on the $r_{O,ex}$ profile. In the titrimetric method the pH of the liquid is controlled around a constant pH setpoint. Thus, in the case of nitrification acid is produced in the first step, and base has to

be added to keep the pH constant. This results in the cumulative base addition curve in Figure 1 (directly proportional to the cumulative proton production).

Table 1. Model used for interpretation of the respirometric and titrimetric data

Comp. (meas., i; substr., k)→ Process (j) ↓	1. X	2. S_S	3. S_O	4. S_{NH}	5. S_{NO2}	6. S_{NO3}	7. H_p	Process rate
1. Growth on S_S	1	$-\frac{1}{Y_H}$	$-\frac{1-Y_H}{Y_H}$	$-i_{XB}$			$\frac{i_{XB}}{14}$	$\mu_{max\,H}\frac{S_S}{K_S+S_S}X$
2. Nitrification step 1	1		$-\frac{3.43-Y_{A1}}{Y_{A1}}$	$-\frac{1}{Y_{A1}}-i_{XB}$	$\frac{1}{Y_{A1}}$		$\frac{i_{XB}}{14}+\frac{1}{7Y_{A1}}$	$\mu_{max\,A1}\frac{S_{NH}}{K_{SA1}+S_{NH}}X$
3. Nitrification step 2	1		$-\frac{1.14-Y_{A2}}{Y_{A2}}$		$-\frac{1}{Y_{A2}}$	$\frac{1}{Y_{A2}}$		$\mu_{max\,A2}\frac{S_{NO2}}{K_{SA2}+S_{NO2}}X$

Results

First, the theoretical identifiability of the two-step nitrification model was studied for separate outputs of $r_{O,ex}$ and Hp data, considering a model structure in which biomass growth did not take place. This means that the first column in Table 1 is not considered (X is constant) and that $i_{XB}=0$. Second, a model structure was studied in which biomass growth was included.

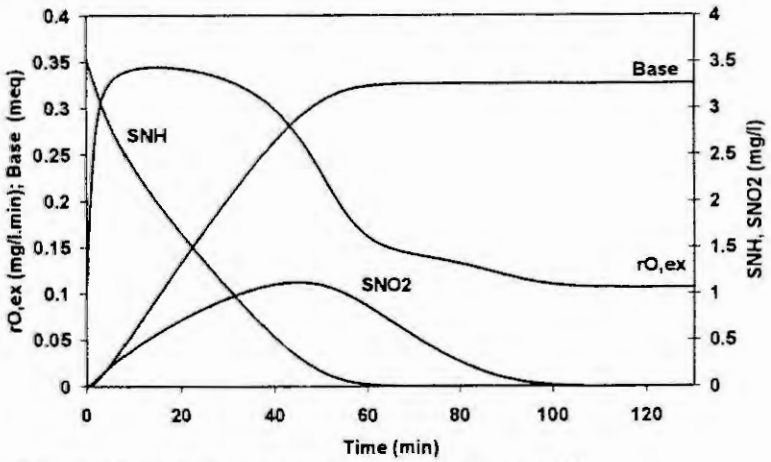


Figure 1 Model example of two-step nitrification [5]

The theoretical identifiability was dealt with separately for the first and the second nitrification step, similar to an earlier study of a double Monod model with two carbon substrates added at t=0 [3]. However, there are two major differences between the two-step nitrification process and this double Monod example: (1) At t=0 the concentration of S_{NO2} is zero. Hence, only information on the kinetics of the first nitrification step is available at t=0; (2) the two nitrification steps are linked in the way that S_{NO2} is produced from the first step, i.e. as long as S_{NH} is present a time varying input of S_{NO2} exists. The study of the theoretical identifiability was therefore approached as follows. First, t=0 was considered and the identifiability of the first step was analysed. Secondly, it was assumed that S_{NH} was completely eliminated from the mixed liquor at a certain point t>0 (in Fig. 1 about t=60min). At this point $S_{NH}=0$ and thus S_{NO2} is no longer produced. However degradation of S_{NO2} still takes place and as a consequence the identifiability of the second nitrification step can be studied. Thus, this is an example in which later observations (t>0) can give complementary information on possibly identifiable parameter combinations [9,11].

The full equations and solutions of the identifiability study are not given here due to their complexity (especially in the case where growth is included) and lack of space. The complete and detailed study will be described elsewhere [8]. Here, only the final results of the study are listed in Table 2. Important to notice is that μ_{maxA1} and X can be separated when growth is considered. Moreover, an extra term including the parameter i_{XB} (incorporation of S_{NH} during growth) appears in the parameter combinations for $S_{NH}(0)$ and K_{SA1}.

It was assumed that measurements of $r_{O,ex}$ and Hp are independent and measured simultaneously in the same system. Hence, the information on identifiable parameter combinations based on $r_{O,ex}$ and Hp data separately, can be combined in the search for possibly new and improved parameter identification with combined data. One should remember that improved identifiability can not be expected for the parameters of the second nitrification step since Hp measurements only give information on the first step. The important result of combining $r_{O,ex}$ and Hp data is that the biomass yield Y_{A1} becomes uniquely identifiable. For a model including growth i_{XB} is assumed known. It appears that Y_{A1} is defined by the ratio between the stoichiometric factors relating $r_{O,ex}$ and the first derivative of Hp to degradation of S_{NH}. The identification of Y_{A1} will result in a unique identification of Y_{A2} as well as long as $S_{NO2}(0)=0$ [8].

Table 2. Schematic overview of the theoretically identifiable parameter combinations for nitrification step 1 and 2, depending on the available measurement(s) and the model structure

Process (j)	Nitrification step 1			Nitrification step 2
Measurement (i) → Model structure ↓	r_O	Hp	r_O + Hp	r_O
No growth	$\frac{3.43-Y_{A1}}{Y_{A1}}\mu_{max\,A1}X$ $(3.43-Y_{A1})K_{SA1}$ $(3.43-Y_{A1})S_{NH}(0)$	$\frac{2}{14}\frac{\mu_{max\,A1}X}{Y_{A1}}$ $\frac{2}{14}K_{SA1}$ $\frac{2}{14}S_{NH}(0)$	$\frac{3.43-Y_{A1}}{Y_{A1}}\mu_{max\,A1}X$ $(3.43-Y_{A1})K_{SA1}$ $(3.43-Y_{A1})S_{NH}(0)$ $\frac{14}{2}(3.43-Y_{A1})$	$\frac{1.14-Y_{A2}}{Y_{A2}}\mu_{max\,A2}X$ $(1.14-Y_{A2})K_{SA2}$ $(1.14-Y_{A2})S_{NO2}(0)$
Growth	$\mu_{max\,A1}$ $\frac{3.43-Y_{A1}}{Y_{A1}}X(0)$ $\frac{3.43-Y_{A1}}{1+i_{XB}\cdot Y_{A1}}K_{SA1}$ $\frac{3.43-Y_{A1}}{1+i_{XB}\cdot Y_{A1}}S_{NH}(0)$	$\mu_{max\,A1}$ $\frac{2+i_{XB}Y_{A1}}{14}\frac{X(0)}{Y_{A1}}$ $\frac{2+i_{XB}Y_{A1}}{14(1+i_{XB}Y_{A1})}K_{SA1}$ $\frac{2+i_{XB}Y_{A1}}{14(1+i_{XB}Y_{A1})}S_{NH}(0)$	$\mu_{max\,A1}$ $\frac{3.43-Y_{A1}}{Y_{A1}}X(0)$ $\frac{3.43-Y_{A1}}{1+i_{XB}Y_{A1}}K_{SA1}$ $\frac{3.43-Y_{A1}}{1+i_{XB}Y_{A1}}S_{NH}(0)$ $\frac{2+i_{XB}Y_{A1}}{14(3.43-Y_{A1})}$	$\mu_{max\,A2}$ $\frac{1.14-Y_{A2}}{Y_{A2}}X(0)$ $(1.14-Y_{A2})K_{SA2}$ $(1.14-Y_{A2})S_{NO2}(0)$

It appears possible to generalise the parameter identifiability results listed in Table 2 based on an ASM1 like stoichiometric matrix (Table 1). It appears that the identifiable parameter combinations can be predicted based on knowledge of the process under study, the measured component and the substrate component that is degraded. This generalisation is illustrated in Table 3, where ν denotes the stoichiometric coefficient, j the process and i the measured component. If i is a component that is consumed (e.g. S_O, S_S) $\nu_{i,j}$ is negative whereas $\nu_{i,j}$ is positive if the measured component is a product (e.g. Hp, X). This is indicated in Table 3 with the factor f (f = +1 or −1). The substrate under study is denoted k and since k is always consumed $\nu_{k,j}$ gets a negative sign. In case two components are measured, the parameter combinations listed in Table 3 still hold but with the additional identifiable parameter combination $\nu_{i(1),j}/\nu_{i(2),j}$ where (1) and (2) indicate the two measured components respectively. The generalisation of Table 3 was confirmed with the identifiable parameter combinations listed in Table 2 [8], but also with examples from literature. E.g. in the study of Holmberg (1982) [7] it was shown that the parameters μ_{max}, K_S, $S_S(0)$, X(0) and Y_H were uniquely identifiable in case S_S and X measurements were available and biomass growth was considered. According to the generalisation (Table 3) and the model (Table 1) this means that i(1)=1, i(2)=2, j=1 and k=2 and the identifiable parameters can be found via :

1) $\mu_{max,j}$, i.e. the growth rate related to process 1 : μ_{maxH}.
2) $f\nu_{i,j}\cdot X$, both i(1) and i(2) can be considered : $\nu_{1,1}\cdot X = X$
3) $-f\nu_{i,j}/\nu_{k,j}\cdot K_j : -\nu_{1,1}/\nu_{2,1}\cdot K_1 \Rightarrow \left(1\cdot\frac{1}{Y_H}\right)K_S = \frac{K_S}{Y_H}$
4) $-f\nu_{i,j}/\nu_{k,j}\cdot S_k(0) : -\nu_{1,1}/\nu_{2,1}\cdot S_2(0) \Rightarrow \frac{S_S(0)}{Y_H}$
5) $f\nu_{i(1),j}/f\nu_{i(2),j} \Leftrightarrow -\nu_{1,1}/\nu_{2,1} \Rightarrow 1\cdot\frac{Y_H}{1} = Y_H$

Table 3. Generalisation of identifiable parameter combinations.

No growth (i_{XB}=0)	Growth
$f.\nu_{i,j}\cdot\mu_{max,j}\cdot X$	$\mu_{max,j}$
$-f.\nu_{i,j}/\nu_{k,j}\cdot K_j$	$f\nu_{i,j}\cdot X(0)$
$-f.\nu_{i,j}/\nu_{k,j}\cdot S_k(0)$	$-f\nu_{i,j}/\nu_{k,j}\cdot K_j$
	$-f\nu_{i,j}/\nu_{k,j}\cdot S_k(0)$

Thus, since the biomass yield Y_H becomes identifiable (step 5) all the parameters μ_{maxH}, K_S, $S_S(0)$, X(0) and Y_H become identifiable by applying the generalisation rules, similar to the results of Holmberg [7].

Discussion

The theoretical identifiability of a two-step nitrification model was studied assuming that respirometric and titrimetric measurements were available. The study was carried out for a model structure that did not include biomass growth and a model structure where biomass growth was taken into account. With respect to parameter

identifiability, the difference between the two model structures was that the no-growth parameter combination including X, Y and μ_{max} can be split up further into μ_{max} on the one hand and a parameter combination including X and Y on the other hand when considering growth. Nitrification kinetic parameters have been estimated in earlier studies [2,10]. In these studies, however, the assumed theoretically identifiable parameter combinations were defined to be the ones related to no growth despite the fact that growth was considered explicitly in the model applied during parameter estimation. Thus, from a theoretical point of view, a wrong approach was taken. However, the experiments considered in these studies were all of short-term character where significant biomass growth is unlikely to take place. Indeed, to be able to practically identify the theoretically identifiable parameters based on a model including growth, the available data must show a significant increase of biomass, e.g. visible in an increase of $r_{O,ex}$. If the data do not reflect a significant growth μ_{max} and X will be correlated in practice.

An important result of this study was that the autotrophic yield, Y_{A1}, becomes uniquely identifiable by combining respirometric and titrimetric data. It is not surprising that a unique identification of the biomass yield requires two kinds of measurements, since the yield in fact relates two measures that link how much biomass is produced per unit of substrate degraded. This was in fact already proven in [7] where combined measurements of biomass and substrate were assumed. The biomass yield becomes identifiable for the combined measurements because an additional parameter combination becomes identifiable compared to the single measurement cases. This additional parameter combination appears to be the ratio of the two stoichiometric factors that relate the respective measured variables to substrate degradation.

Finally and most substantially, it was proven and illustrated how it is possible to generalise the theoretical parameter identifiability based on an ASM1-like stoichiometric matrix. The identifiable parameter combinations can be predicted directly based on knowledge of the process under study, the measured component and the substrate component that is degraded. This generalisation is a very powerful tool since it reduces the time-consuming task of assessing the theoretical identifiability of process models described by Monod growth kinetics in an ASM1-like representation.

Conclusions

In this study the theoretical identifiability of the two-step nitrification model was studied considering respirometric and titrimetric data. The result of including biomass growth in the model was, not surprisingly, that the parameter μ_{max} could be identified separately. However, more important was that the parameter identification improves when combined respirometric and titrimetric data are available since the autotrophic yield becomes uniquely identifiable. Finally, the results were generalised and it was shown how identifiable parameter combinations could be obtained directly from an ASM1-like matrix representing the model under study.

Literature

1. Bourrel, S.V., Babary, J.P., Julien, S., Nihtilä, M.T. and Dochain, D., Modelling and identification of a fixed-bed denitrification bioreactor, System Analysis Modelling Simulation (SAMS), 30 (1998), 289-309.
2. Brouwer, H., Klapwijk, A. and Keesman, K.J., Identification of activated sludge and wastewater characteristics using respirometric batch experiments, Water Res., 32 (1998), 1240 - 1254.
3. Dochain, D., Vanrolleghem, P.A. and Van Daele, M., Structural identifiability of biokinetic models of activated sludge respiration, Water Res., 29 (1995), 2571-2578
4. Gernaey, K., Vanrolleghem, P.A. and Verstaete, W., On-line estimation of Nitrosomonas kinetic parameters in activated sludge samples using titration in-sensor-experiments, Water Res., 32 (1998), 71 - 80.
5. Gernaey, K., Petersen, B., Ottoy, J.P. and Vanrolleghem, P.A., Activated sludge monitoring with combined respirometric – titrimetric measurements, Submitted to Water Res. (1999).
6. Henze, M., Grady, C.P.L. Jr., Gujer, W., Marais, G.v.R. and Matsuo, T., Activated sludge model No. 1, IAWQ Scientific and technical Report No. 1, London, 1987.
7. Holmberg, A., On the practical identifiability of microbial growth models incorporating Michaelis-Menten type nonlinearities, Mathematical Biosciences, 62 (1982), 23-43.
8. Petersen, B., Optimal experimental design for wastewater and sludge characterisation, Ph.D. thesis. Biomath department, University Gent, B-9000 Belgium. In preparation, 2000.
9. Pohjanpalo, H., System identifiability based on the power series expansion of the solution, Mathematical Biosciences, 41 (1978), 21-33.
10. Spanjers, H. and Vanrolleghem, P.A., Respirometry as a tool for rapid characterization of wastewater and activated sludge, Water Sci. Technol., 31(2) (1995), 105 - 114.
11. Walter, E., Identifiability of state space models, Springer, Berlin, 1982.

SIMULATION OF WASTEWATER TREATMENT PLANTS

M.N. Pons, O. Potier, E. Olmos, J. Fougea, N. Roche, C. Prost
Laboratoire des Sciences du Génie Chimique, CNRS-ENSIC-INPL
1, rue Grandville, BP 451, F-54001 Nancy cedex, France

Abstract. A dynamic simulator of a full-scale wastewater plant have been developed taking into consideration the hydrodynamics, the bioreactions and the physical phenomena. It is used as a platform to study different control strategies and scenarios of the plant revamping.

Introduction.

Wastewater treatment plants are non-linear systems subject to large perturbations in flow and load, together with uncertainties concerning the composition of the incoming wastewater. Nevertheless these plants have to be operated continuously, meeting stricter and stricter regulations. It is therefore difficult to test on-line new control strategies, to investigate the modifications in a plant or to train the operators. Simulators can be very useful, if they represent sufficiently well the reality. For that purpose it is necessary to have a correct description of the hydrodynamics, as the plants are generally of considerable size, of the biological reactions, the physical phenomena (settlement, flocculation, de-watering, etc.), and to have access to realistic influent data files (composition and flow rate). This paper describes the development of an in-house PC-based dynamic simulator for a full-scale wastewater plant.

Modelling the plant

The simulator is based on the actual full-scale plant of Nancy-Maxeville (350 000 eq. inh.) in France, that is a carbon removal process by activated sludge. The plant has three lines working in parallel and one line is simulated.

General process description

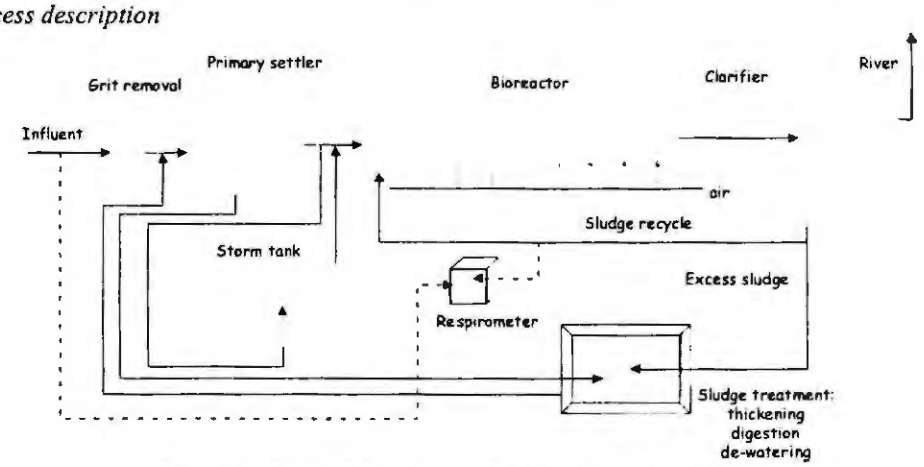


Figure 1: General flowsheet of the wastewater simulator

The flowsheet is represented in Figure 1. Most of the insoluble pollution is removed by two parallel primary settlers (lamellar type) of 1000 m^3. If the incoming flow rate exceeds some limit, part of the wastewater can be stored in a storm tank (volume 4500 m^3), the content of which is later treated. The bioreactor is a channel of volume 3300 m^3 and length 100 m, with gas diffusers for aeration. The circular clarifier has a volume of 5000 m^3. The sludge treatment (thickening, anaerobic digestion, de-watering) is not modeled but some residual wastewater coming from this section is injected between the grit removal and the primary settler. Constant flow and composition are assumed for this residual wastewater. The respirometer is operated batchwise: using recycled sludge and influent wastewater, it enables to detect toxics present in the incoming wastewater.

Bioreactions.

The IAWQ Activated Sludge Model N° 1 [1] was chosen to simulate the biological process taking place in the reactor. The removal of carbon and nitrogen components by heterotrophs and autotrophs is considered. The pollution is divided into a soluble fraction and an insoluble fraction, each of them having a biodegradable fraction and a non-biodegradable fraction. The biological pathway is summarized in Figure 2. Classical inhibition functions for the growth rate have been added and the death rates are also function of the toxics concentration, that can be biodegraded.

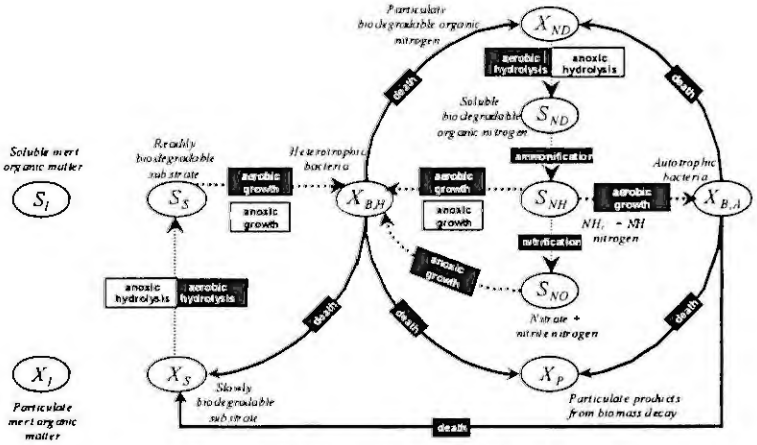


Figure 2: Schematic representation of the biopathway

Hydrodynamics

Due to the size of each sub-process (primary settler, bioreactor, clarifier) a careful modeling of the hydrodynamical behavior should be undertaken. Two approaches can be used: a fluid mechanics approach based on Navier-Stokes equations (CFD) and a systemic approach based on combinations of well-mixed reactors. Although CFD enables a very precise description, it requires a very high computer power. A mesh of more than 500 000 cells is necessary to represent the 3000 m^3 aerated channel under Fluent™5. Figure 3a shows a 8m x 4m x 4m channel with two ramps of gas diffusers on the bottom. 12hrs of calculation were necessary to obtain the convergence of the velocity field, in absence of bioreaction, on a Pentium 450 MHz. A section of the velocity field is shown in Figure 3b. It can be shown that the recirculation loops are function of the influent flow rate and on the gas flow rate.

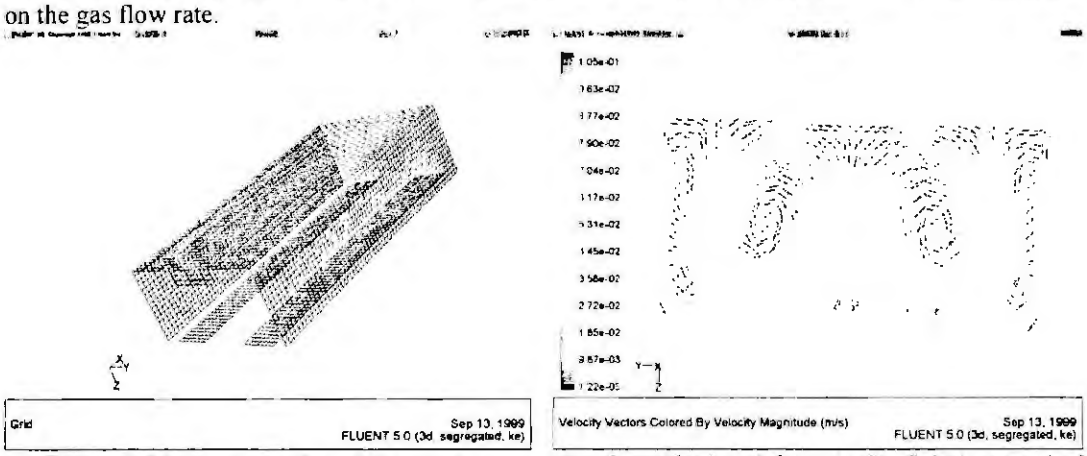


Figure 3: Mesh of a section of the short rectangular channel (a) and the velocity field in a vertical vectical section (b)

A more useful model for our dynamic simulation purpose is obtained by representing the bioreactor by a series of units as shown in Figure 4. This approach is classical in chemical engineering. The number of units and the back-mixing flow rate are adjusted from residence time distributions [2], obtained by injections of a inert tracer (a lithium salt for example) in the reactor. The back-mixing flow is a function of the influent flow rate and of the gas flowrate [2]. It is not necessarily the same for all units. For example, in the case of anoxic zone, it is considerably reduced.

Similarly 2-dimensional models were selected for the settler and the clarifier (Figure 5). The well-mixed cells are rectangular for the lamellar primary settler and concentric crowns for the clarifier [3, 4]. Their number and size have been deduced from residence time distributions on the real process units.

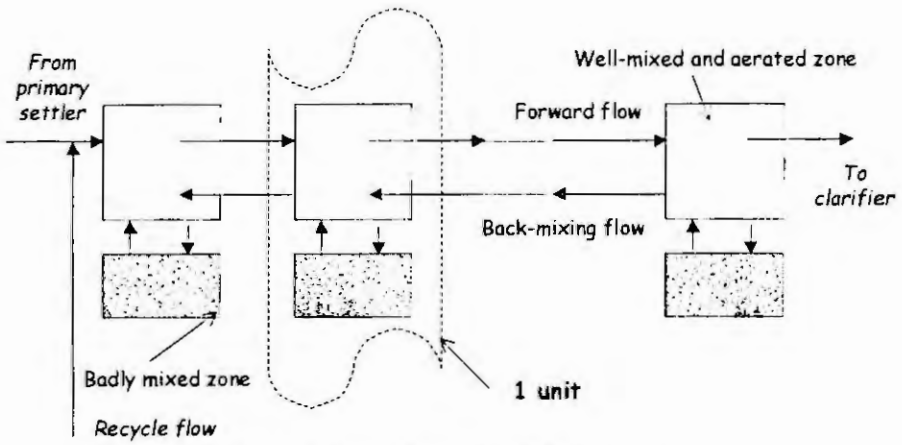


Figure 4: Systemic model of the bioreactor

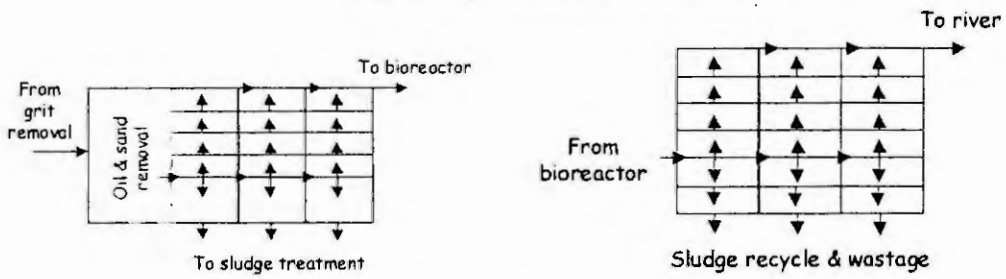


Figure 5: Systemic model of the lamellar primary settler (a) and of the clarifier (b)

Settling characteristics

The settling velocity in the primary settler is described by the Stokes' equation. The double-exponential settling velocity model proposed by Takács *et al.* [5] has been selected to describe the behaviour of the sludge in the clarifier.

Influent datafiles

Two types of influent data files are available: the files from the benchmark proposed by COST 624 (http://www.ensic.u-nancy.fr/COSTWWTP) and periodical functions, the characteristics of which have been deduced on the Nancy-Maxeville plant [5]. The COST 624 data files are dependent of the weather conditions.

The average dry weather flowrate treated on one line of the Nancy-Maxeville plant is about 600 m^3/h with an average soluble COD of 180 mg/l and insoluble COD of 130 mg/l. Figure 6 presents some of the experimental data collected on the plant concerning the nitrogen components. The concentrations are correlated with the time of the day but also with the weather conditions: this is particularly noticeable on the nitrate data in Figure 6b: nitrate concentration in the influent increases significantly after a rain event, due to the storage of rain water in storm tanks in the sewage system and some oxygenation of the influent. The wastewater from the de-watering unit represents a small fraction of the total flow rate (≈10%) but the pollution is very concentrated (≈1,5 g/l for total DCO).

(a) (b)

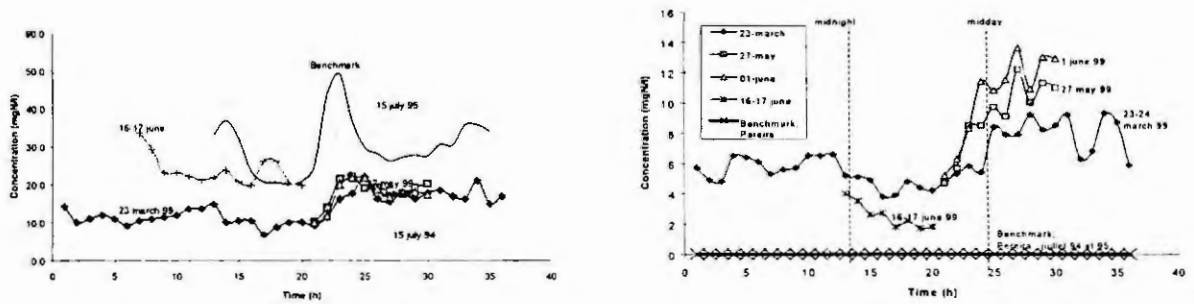


Figure 6: Nitrogen component experimental data: (a) ammonia and nitrates (b) for various days and comparison with the COST624 benchmark values (dry weather)

Discussion and conclusions.

The different models have been implemented in FORTRAN and the simulator can run on a PC. No special effort toward user-friendliness and graphical interface was made however. It is more oriented toward research than training. Its key features are in the hydrodynamical approach and this part of the models could probably be implemented on an existing commercial software.

The software is currently used to investigate the effect of the de-watering wastewater, as it contributes significantly to the pollution introduced in the bioreactor, in spite of the small flow rate, to optimize the operation of the respirometer that is used to detect toxics in the influent [6], as well as the strategy for discharging the storm water back into the bioreactor. Figure 7 presents an example of use of the software with the effect of the time of release (middle of the week or during a week-end) of a toxic in the influent on the maximal oxygen uptake rate of the sludge as measured in the respirometer.

Further work is also going on to use results from CFD simulations to estimate the hydrodynamical parameters of the bioreactor model, to avoid to make a large number of residence time distribution experiments.

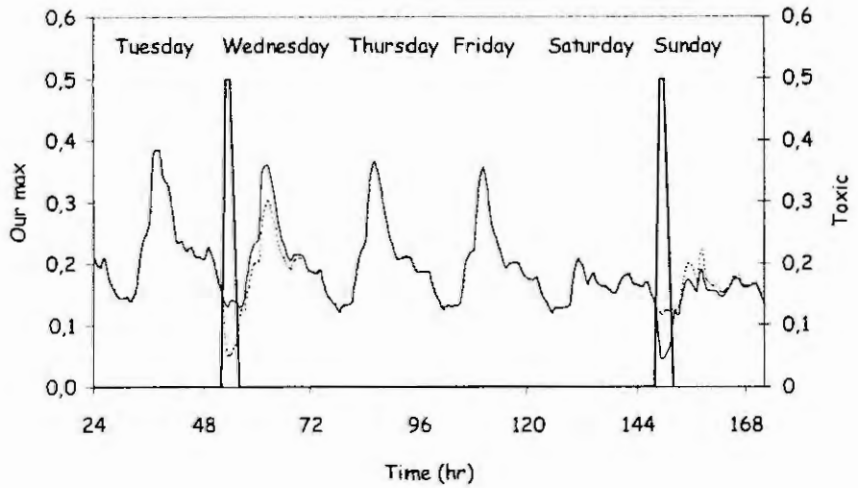


Figure 7: Effect of the time of toxic release {(---) Wednesday and (—) Sunday } on the response of the respirometer

Acknowledgements.

The authors are thankful to the Communauté Urbaine du Grand Nancy for its help.

References.

1. Henze, M., Grady, Jr C.P.L., Gujer, W., Marais, G.v.R. and Matsuo, T., Activated sludge model n°1, IAWQ Scientific and Technical Report n°1, IAWQ, London (1986).
2. Potier, O., Pons, M.N., Roche, N., Leclerc, J.P., Galdemas, L. and Prost, C., Etude de l'hydrodynamique d'un réacteur canal à boues activées en régime variable, Récents Progrès en Génie des Procédés, Lavoisier, 12, 61 (1998) 367-372.
3. Pereira, L., Modélisation et simulation d'une station d'épuration des eaux usées urbaines par boues activées, PhD Thesis, INPL, Nancy, France (1996).
4. Pons, M.N., Roche, N., Potier, O. and C. Prost, Modélisation de l'hydrodynamique de décanteurs primaires et secondaires de stations d'épuration des eaux usées par boues activées, Récents Progrès en Génie des Procédés, Lavoisier, 12, 61 (1998) 115-120.
5. Takács, I., Patry, G.G. and, Nolasco, D., A dynamic model of the clarification thickening process, Water Research, 25 (1991) 1263-1271.
6. Pons, M.N., Roche, N., Blanc, C., Potier, O., Prost, C., Nieddu P. and Cécile, J.L., Detection of critical situations in wastewater treatment plants, Proceedings ECB9, under press (1999).

MODELLING AND ESTIMATION OF SPECIFIC GROWTH RATE FOR DENITRIFICATION OF WASTEWATER

M. Nadri [1], H. Hammouri [1] and M. Fick
[1]Université Claude Bernard Lyon I
L.A.G.E.P. Bât 308G. 69622 villeurbanne

Abstract
The aim of this paper consists in modelling and identifying biological parameters of denitrification bioreactors and essentially to develop some methods for on-line estimation of specific growth rate of cells. The proposed model assume that the kinetics are described using Monod expressions with substrate limitation. First, using batch experimental data, we identify the parameters of the model. Afterwards, a nonlinear estimator of specific growth rate is proposed. Finally, the performance of the proposed estimator is tested in simulation
Key-Words: denitrification, nonlinear systems, estimator, anaerobic fermentation

I Introduction

In this paper, we study a particular case of bioprocesses concerning a biological denitrification processes. This process is frequently considered to be the most feasible and cost-effective processes available for nitrogen removal. The presence of nitrogen in the aquatic environment accelerates entrophication and leads to worse water-quality. However, the regulation of the nitrogen (nitrate and nitrite) content of drinkable water and their control tends to be more stringent.
In this work, denitrification was carried out for a pure culture of Pseudomonas denitrificant and was assumed to occur with the consecutive reduction of nitrate to nitrite and nitrite to nitrogen gas without accumulation of intermediate gaseous products .
The objective of this work is not only to develop dynamics model that adequately describes the principal of biological denitrification, but essentially to develop some methods for the on-line estimation of specific growth rate (μ) from biomass and consumption of carbon. At last, we show that this estimation can be used for the estimation of content of nitrate and nitrite. This paper is organized as follows: In second section, we give the description of the bioreactor and we validate a mathematical model from experimental data. In third section, we design a simple estimator permitting to estimate the specific growth rates.

II Dynamic model of denitrification bioreactor

Biological denitrification can convert all the nitrate in wastewater treatment into gaseous nitrogen. Many microorganisms can denitrify under anaerobic conditions (e.g. Pseudomonas, Alcaligenes, Bacillus...). In our case, we have select Pseudomonas denitrifians. The final electron acceptor is nitrate instead of oxygen. The carbon substrate is used as the electron donor. There is various carbon sources, acetic acid was selected as carbon substrate in this study.
The denitrification process considered is a culture fermentor (volume 1L) with complete monitoring and control instrumentation for batch experiments. The pH measured with a glass-electrode. The fermentor parameters (agitation, temperature, pH) were automatically controlled on to set-point values. The nitrite, nitrate and acetic acid concentrations are measured by chromatography.
Denitrification is defined as the reduction, by anaerobic bacterias, of both ionic nitrogen oxides (nitrate and nitrite) to gaseous nitrogen products by using organic substrates (electron donor) with the production of intermediate compounds. From the four actual steps of the reduction, we assume that the reduction rates of the two gaseous nitrogen oxides are sufficiently faster than the reduction rates of nitrate NO_3^- and nitrite NO_2^-, then, we consider only the following sequence:

$$NO_3^- \longrightarrow NO_2^- \longrightarrow N_2$$

Several models are given in the literature for pure founders ([8],[9]).
A mass balance dynamical model for Batch fermentation is given by:

$$\begin{cases} \dot{x} = (\mu_3 + \mu_2)s_c x - k_d x \\ \dot{s_2} = (\dfrac{\alpha}{y_3}\mu_3 - \dfrac{1}{y_2}\mu_2)s_c x - k_{m3} x \\ \dot{s_3} = -\dfrac{1}{y_3}\mu_3 s_c x - k_{m2} x \\ \dot{s_c} = (\dfrac{-y_{c3}}{y_3}\mu_3 - \dfrac{y_{c2}}{y_2}\mu_2)s_c x - k_{mc} x \end{cases} \quad (1.\ 1)$$

where μ_i corresponds to specific growth rates given by Monod law :

$$\begin{cases} \mu_3 = \dfrac{\mu_{3max} \cdot s_3}{s_3 + k_3} \cdot \dfrac{k_{I3}}{s_2 + k_{I3}} \cdot \dfrac{1}{s_c + k_{c3}} \\ \mu_2 = \dfrac{\mu_{2max} \cdot s_2}{s_2 + k_2} \cdot \dfrac{k_{I2}}{s_3 + k_{I2}} \cdot \dfrac{1}{s_c + k_{c2}} \end{cases} \quad (1.\ 2)$$

Where μ_{imax} are the maximum specific growth rates for the cells growth by reducing nitrate and nitrite. $k_i (i = 2, 3, c)$ are, respectively, the nitrite, nitrate and carbon half saturation constants and k_{Ii} are the nitrate and the nitrite inhibition constants.
A preliminary sensitivity analysis showed that the output of the model is relatively insensitive to the changes in the saturation constants $k_i (i = 2, 3, c)$ of the Monod model. Consequently, constant values (calculated from experimental data) without estimation were chosen from them. Therefore four parameters are identified: μ_{imax}, $k_i (i = 2, 3)$ and y_i:

$\mu_{2max} = 0.065h - 1$, $\mu_{3max} = 0.28h^{-1}$
$k_2 = 0.02gL^{-1}$, $k_3 = 0.014gL^{-1}$, $_{c2} = 0.05gL^{-1}$, $k_{c3} = 0.05gL^{-1}$, $k_{I2} = 0.05gL^{-1}$, $k_{I3} = 1gL^{-1}$
$y_{c3} = 0.26$, $y_{c2} = 0.82$, $y_2 = 0.66$, $y_3 = 0.23$, $\alpha = 0.61$

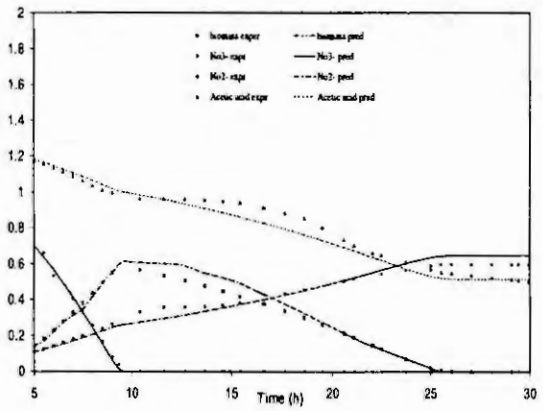


Figure2: Experimental and theoretical profiles of biomass, nitrate, nitrite and acid acetic concentration for Batch culture of P. denitrificant

In the sequel,the model(1. 1) will be used to the mathematical model as simulator permitting to validate our estimator methodology.

III Estimation of Kinetic rate in bioreactors

III-1 The reduced model and observer syntheses

Since, we only deal to estimate the kinetic rates, the considered model can be reduced to the first and the end differential equations of (1. 1).In the sequel, $\mu_x(t)$, and $\mu_s(t)$ are considered as unknown variables and twice differential with bounded derivatives. The reduced model takes the form:

$$\begin{cases} \dot{x} = \mu_x x s_c - k_d x - Dx \\ \dot{\mu_x} = \tilde{\mu_x} \\ \dot{\tilde{\mu_x}} = \varepsilon_x(x,t)\dot{s_c} = \mu_c x s_c - k_m x - D(s_c - s_{cin}) \\ \dot{\mu_s} = \tilde{\mu_s} \\ \dot{\tilde{\mu_s}} = \varepsilon_s(s_c,t) \\ y(t) = \begin{pmatrix} y_1(t) \\ y_2(t) \end{pmatrix} = \begin{pmatrix} x(t) \\ s_c(t) \end{pmatrix} \end{cases} \quad (1.\ 3)$$

where $y_1(t)$ and $y_2(t)$ are the on-line output measurements.
And using the following notations:

$$\begin{cases} \mu_x(t) = \mu_3(t) + \mu_2(t) \\ \mu_s(t) = -\dfrac{y_{c3}}{y_3}\mu_3(t) - \dfrac{y_{c2}}{y_2}\mu_2(t) \end{cases}$$

and $z^1 = \begin{pmatrix} x \\ \mu_x \\ \tilde{\mu_x} \end{pmatrix} = \begin{pmatrix} z_1 \\ z_2 \\ z_3 \end{pmatrix}$ $z^2 = \begin{pmatrix} s_c \\ \mu_s \\ \tilde{\mu_s} \end{pmatrix} = \begin{pmatrix} z_4 \\ z_5 \\ z_6 \end{pmatrix}$ and $\varepsilon^1 = \begin{pmatrix} 0 \\ 0 \\ \tilde{\varepsilon_x} \end{pmatrix}$, $\varepsilon^2 = \begin{pmatrix} 0 \\ 0 \\ \tilde{\varepsilon_s} \end{pmatrix}$

System (1. 3) takes the form:

$$\begin{cases} \dot{z^1} = F^1(u,z) + \varepsilon^1(t) \\ \dot{z^2} = F^2(u,z) + \varepsilon^2(t) \\ y = \begin{pmatrix} x \\ s_c \end{pmatrix} = \begin{pmatrix} cz^1 \\ cz^2 \end{pmatrix} \end{cases} \tag{1. 4}$$

Theorem : *The following system:*

$$\begin{cases} \dot{\hat{z}}^1 = F^1(u,\hat{z}) - \Lambda^1(u,\hat{z}) S_\theta^{-1} C^T (Cz^1 - y) \\ \dot{\hat{z}}^2 = F^2(u,\hat{z}) - \Lambda^2(u,\hat{z}) S_\theta^{-1} C^T (Cz^2 - y) \end{cases} \tag{1. 5}$$

is an exponential estimator for (1. 4):

$$\|\hat{z}(t) - z(t)\| \leq \lambda_\theta \exp(-\mu_\theta t) + \gamma_\theta \varepsilon$$

Where $\varepsilon = sup_{t \geq 0} \|\varepsilon(t)\|$, $\lambda_\theta \geq 0$, $\mu_\theta \geq 0$, $\gamma_\theta \geq 0$, $\varepsilon = \begin{bmatrix} \varepsilon^1 \\ \varepsilon^2 \end{bmatrix}$,

$\Lambda^1(s,z) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & x & 0 \\ 0 & 0 & x \end{bmatrix}$, $\Lambda^2(s,z) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & s_c & 0 \\ 0 & 0 & s_c \end{bmatrix}$ and $S_\theta^{-1} C^T = \begin{bmatrix} 3\theta \\ 3\theta^2 \\ \theta^3 \end{bmatrix}$ where $\theta > 0$ The proof of this result can be obtained in a similar way as in [3]

III-2 Estimation of the μ_i

$\hat{\mu_2}$ and $\hat{\mu_3}$ have been deducted from $\hat{\mu_x}$ and $\hat{\mu_s}$:

$$\begin{aligned} \hat{\mu_2} &= \frac{\hat{\mu_s} - Y_3 \hat{\mu_s}}{Y_2 - Y_3} \\ \hat{\mu_3} &= \hat{\mu_x} - \hat{\mu_2} \end{aligned} \tag{1. 6}$$

The following figures illustrate the performances of observator (1. 5) in simulation in the batch and continuous modes.

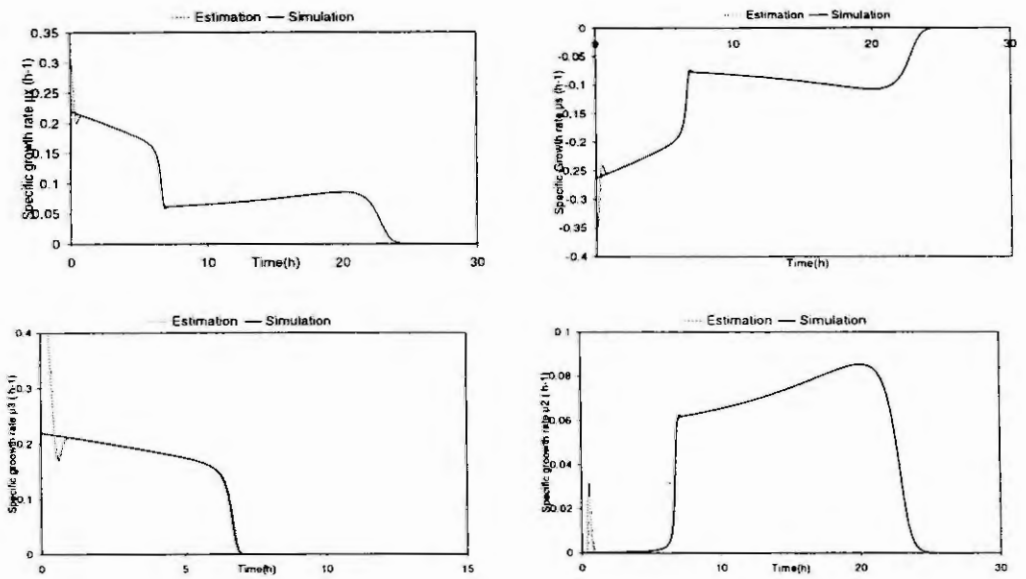


fig.3: Estimation of μ_x, μ_s and μ_3, μ_2 in batch mode

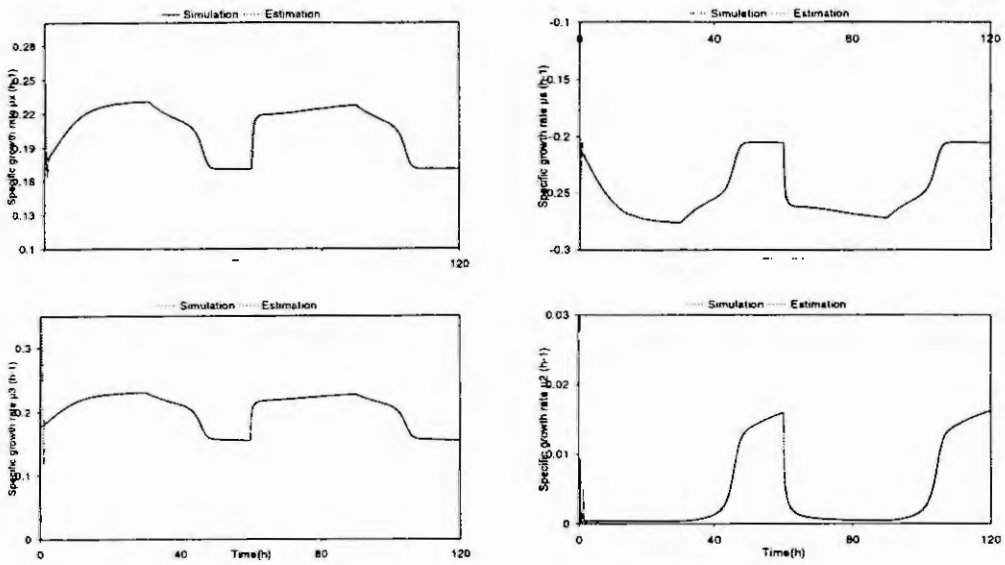


fig.4: Estimation of μ_x, μ_s and μ_3, μ_2 in continuous mode

Two sets of simulation results are given in fig.3 and fig.4 and they respectively correspond to the batch and continuous modes. We have reported on these figures the comparison of μ_x, μ_s, μ_3, μ_2 estimates with the truth values (computed using equation (1. 2)). The value of θ was chosen equal to 5 in the batch mode and to 3 in the continuous mode.
We remark the good agreement between estimated and simulated curves. The obtaiued results are quite satisfactory.

IV Conclusion

The identification of biological parameters is often a complex problem due to the fact that on-line measurements of state variables are neither reliable nor very accurate, and not available for uitrate and nitrite concentration; furthermore a mathematical model derived from mass balances with estimation of specific growth rate. The main characteristic of the proposed estimators lies in the ease with which they are implemented and calibrated. Indeed, the gain of these estimators does not necessitate the resolution of any equatiou and is explicitly given. Moreover, its tuuing is achieved through the choice of a siugle constant parameter.
Simulatiou results were giveu and they demoustrated the good performances of the giveu estimators in coping with non-linearities and in tracking the parameters abrupt variations.

References

[1] J.s Alreida, S.M. Julio: Biotech bioeng vol 46 p 194-201 1995; Nitrite inhibition of denitrification by Pseudomonas fluonescens.
[2] G. Bastin et . Dochain: 1985 à liege; la modélisation et l'automatisation des procédés de fermentation.
[3] G.Boruard and H.Hammouri: A high gain observer for a class of uniformly observable systems, IEEE CDC, Brighton, GB, (1991).
[4] Busawou, K.; Farza, M.; Hammouri, H.: A simple observer for a class of nonlinear systems. Applied Mathematics Letters 11 (1998) 27-31.
[5] BoXu and S. Olof : J. of Fermentation and bioeugi V 82 N1 p 56-60 199; Modeling of Nitrite Accumulation by the Denitrifying Bacterium Pseudomonas Stutzeri.
[6] H. Costantin: thése 1995 INPL; La biodénitrificatiou d'un effluent iudustriel fortement chargé, Etudes cinétiques, conception d'un réacteur a lit fluidisé et modélisation.
[7] M. Kornaros et all: water environoment Reserch V 68 N 5 Denitrificans under groeth conditions limited by carbou and/or nitrate uitrite.
[8] C.Pevtavin: thése 1995 INPL ; Réduction des nitrates et des nitrites en N_2 par Pseudomonas Stutzeri: etudes cinétiques, modélisation et simulation d'un système dénitrifiant.
[9] J. H. Wang, B.C. Baltzis: Biotech bioeng V 47,P 26-41 1995; Fundamental Denitricatiou Kinetic Studies whith Pseudomonas denitrificans.

KINETIC MODELING OF AEROBIC DENITRIFICATION BY MICROVIRGULA AERODENITRIFICANS

J. Harmand[1], D. Patureau[1], *C. Armaing[1], I. Queinnec[2], J. P. Steyer[1]
[1]Laboratoire de Biotechnologie de l'Environnement,
Institut National de la Recherche Agronomique,
Avenue des étangs, 11100 Narbonne, France.
[2]Laboratoire d'Analyse et d'Architecture des Systèmes,
Centre National de la Recherche Scientifique,
7, Avenue du Colonel Roche, 31077 Toulouse Cedex, France.

Abstract. The micro-organism *Microvirgula aerodenitrificans* has a particular behavior with respect to the oxygen. Indeed, it is able to simultaneously reduce nitrate and oxygen. The aim of this study is to model the growth rate of this bacteria in order to analyze the possibility to maintain its activity within a complex ecosystem. Two models are identified according to the specific conditions of aeration (anoxic or aerobic conditions). The identified models are presented and confronted to experimental data. Some conclusions and perspectives are then drawn.

Introduction

Nitrogen compounds such as ammonium and N-oxides are now considered as an important and crucial environmental problem. In natural ecosystems, these compounds are normally removed microbiologically in two steps. Ammonium is first oxidized by aerobic, autotrophic microorganisms into nitrite and/or nitrate. In the second step, the N-oxides are reduced into dinitrogen gas by anoxic, generally heterotrophic microorganisms.

The conventional biological nitrogen removal plants are also based on either space or time separation of these two phases. However, for the last ten years, many studies have reported existence of atypic strains that are able of heterotrophic nitrification [11], aerobic denitrification [12], [7], anaerobic conversion of ammonium into nitrogen [5], anoxic reduction of N-oxides into nitrous oxide or nitrogen by nitrifiers [1], [3]. Existence of these atypic behaviours might be an attractive alternative for wastewater treatment plants. In this way, *Microvirgula aerodenitrificans*, has been isolated in our laboratory. It has recently been established that this strain can stably maintain the ability to co-respire oxygen and N-oxides and to produce simultaneously dinitrogen gas [8] and [9].

Several additional studies have pointed out the feasability of combining this pure strain with a complex nitrifying microflora in a single aerobic continuous or sequencing batch reactor for nitrogen removal [10]. Moreover, in order to monitor the pure strain in this ecosystem, molecular tools such as fluorescent *in situ* hybridization with rRNA-targeted nucleic acid probe has been used [4]. However, some problems remained, such as long term strain maintenance or aerobic denitrifying activity maintenance. Then, the idea to model this strain in order to better understand its specific behavior has grown. Indeed, mathematical modeling of this strain could be more valuable to estimate the potential application of such microorganism, to predict its behavior in a complex system and to answer these maintenance questions. In this study, the kinetic modeling of this strain is reported. Simulation results are compared with experimental data before some perspectives are drawn.

Structural analysis

Structure of the mass balance model

In this study, the strain *Microvirgula aerodenitrificans* has been modeled both in complete anoxic and aerobic batch conditions. In both cases, the structure of the model used was defined as in (1) in which X is the pure strain concentration, S_1 the acetate concentration S_2 the nitrate ($N\text{-}NO_3^-$) concentration and S_3 the ammoniac ($N\text{-}NH_4^+$) concentration. The function μ is the specific growth rate of the biomass while $Y_{X/Si}$ are the yield consumption coefficients associated with each of the substrates.

---

* *Actually with the "Autoroute du Sud de la France" company.*

$$\begin{cases} \dfrac{dX}{dt} = \mu X \\ \dfrac{dS_1}{dt} = -\dfrac{\mu}{Y_{X/S1}} X \\ \dfrac{dS_2}{dt} = -\dfrac{\mu}{Y_{X/S2}} X \\ \dfrac{dS_3}{dt} = -\dfrac{\mu}{Y_{X/S3}} X \end{cases} \quad (1)$$

Structural analysis of the growth rate function

During experiments, it was noted that growth rate of the biomass was limited by the three substrates while inhibited by high concentrations of acetate (S_1). For the limiting influence of the substrates, Monod structures were used while the inhibition by the acetate was modeled using the expression proposed in [6], that is :

$$\mu_{inhib} = 1 - (\frac{S_1}{K_1})^{\alpha} \quad (2)$$

with α a real scalar and K_1 an inhibition constant to be identified.

Furthermore, using experimental data, it was pointed out that the maximum growth rate is a polynomial function of the pH. As a consequence, the final growth rate structure was chosen as :

$$\begin{aligned} \mu &= \mu_{max}(pH)\mu(S_1)\mu(S_2)\mu(S_3)\mu_{inhib} \\ &= \mu_{max}(pH)\frac{S_1}{S_1 + K_{S1}}\frac{S_2}{S_2 + K_{S2}}\frac{S_3}{S_3 + K_{S3}}\left(1 - (\frac{S_1}{K_1})^{\alpha}\right) \end{aligned} \quad (3)$$

where $\mu_{max}(\text{pH}) = a\text{pH}^4 + b\text{pH}^3 + c\text{pH}^2 + d\text{pH} + pH$ with a, b, c and d some real scalars to be identified.

The expression of the growth rate given in (3) is valid in anoxic conditions while, in aerobic conditions, the limitation term by the nitrate disappears. Indeed, if nitrate is not present, oxygen is used instead by *Microvirgula aerodenitrificans* for its growth. Furthermore, in aerobic conditions, the experiments were not performed at different pH. As a consequence, the pH influence under these conditions could not be taken into account. Finally, the following expressions were used :

Under anoxic conditions $\mu = \mu_{max}(pH)\frac{S_1}{S_1 + K_{S1}}\frac{S_2}{S_2 + K_{S2}}\frac{S_3}{S_3 + K_{S3}}\left(1 - (\frac{S_1}{K_1})^{\alpha}\right)$

Under aerobic conditions $\mu = \mu_{max}\frac{S_1}{S_1 + K_{S_1}}\frac{S_3}{S_3 + K_{S_3}}\left(1 - (\frac{S_1}{K_1})^{\alpha}\right)$

Identification of model parameters

Before identifying the model parameters, the study of the theoretical identifiability of these parameters was realized [2]. It was concluded that, under specific operating conditions - both in aerobic and in anoxic conditions - all parameters of the models could be identified. For anoxic conditions, the yield coefficients were computed using the simple formula $Y_{X/Si} = dX/dS_i$. However, in aerobic conditions, due to the fact that nitrate is less consumed in the presence of oxygen, a function of oxygen was used :

$$Y_{X/S_2} = \sigma f(o_2) \quad (4)$$

For identifying the other model parameters, a standard nonlinear optimization algorithm was used. The criterion to be minimized was chosen to be the Sum of Square Error (SSE). The identification of the model parameters led to the following values :

Under anoxic conditions : $\mu_{\max}(\text{pH}=6.5) = 0.25\ \text{h}^{-1}$, $K_{S1} = 1$ mg/l, $K_{S2} = 5$ mg/l, $K_{S3} = 100$ mg/l, $K_1 = 35000$ mg/l, $\alpha = 1$.

Using the data plotted in Figure 1 and the function expression given in (3) allow us to compute the parameters as :

$$\mu_{max} = -0.39\ \text{pH}^4 + 10.89\ \text{pH}^3 - 112.78\ \text{pH}^2 + 518.34\ \text{pH} - 891.55 \quad (5)$$

Table 1 : Values of μ_{max} as a function of the pH under anoxic conditions

PH	6	6.5	7	7.5	8
μ_{max}	0.1	0.25	0.25	0.3	0.01

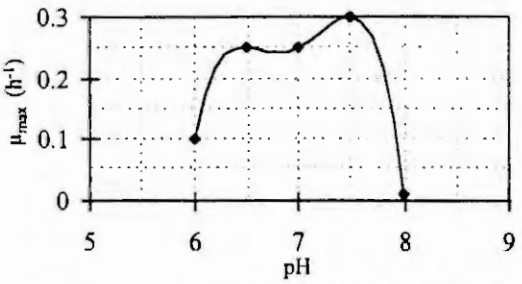
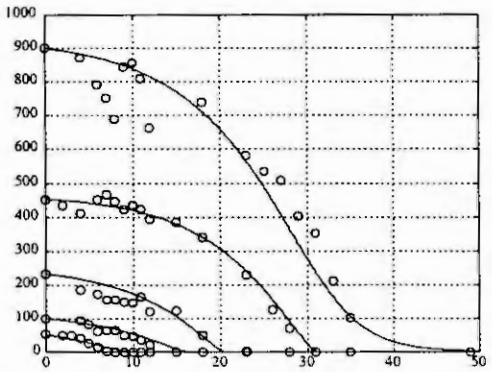


***Figure 1** : μ_{max} as a function of the pH under anoxic conditions*

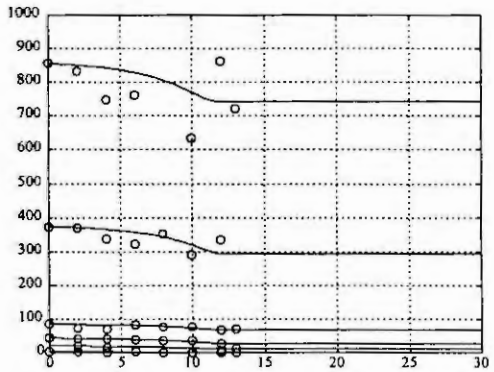
Under aerobic conditions : μ_{max}=0.4 h^{-1}, K_{S1}=80 mg/l, K_{S3}=5 mg/l, K_1=35000 mg/l, α=1.

Experimental data versus simulation results

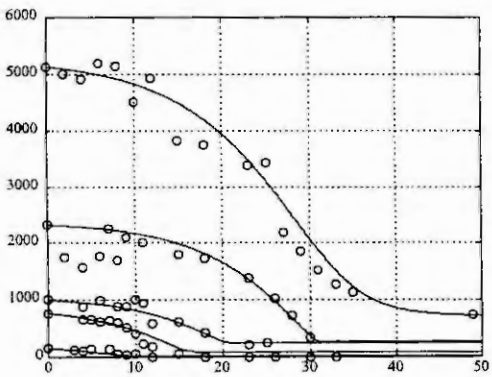
In order to evaluate these models, the results of simulations were confronted to the experimental data. Due to the large number of different experiments performed (more than 30 batch experiments), it is not possible to present all validation curves. The reader can refer to [2] for further details. The most significant results are presented hereafter for different initial conditions in both anoxic and aerobic conditions.



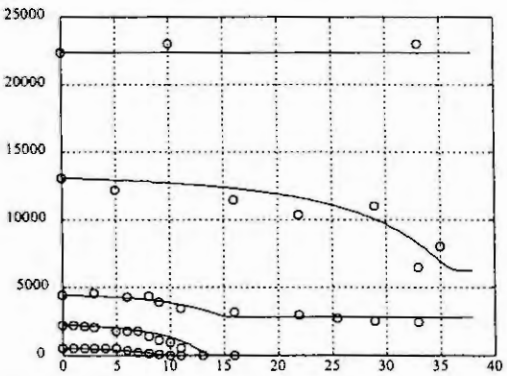
*(**Figure 2** a) N-NO_3^- (mg/l) vs. Time (h) for different initial conditions under anoxic conditions*



*(**Figure 2** b) N-NO_3^- (mg/l) vs. Time (h) for different initial conditions under aerobic conditions*



*(**Figure 2** c) Acetate (mg/l) vs. Time (h) for different initial conditions under anoxic conditions*



*(**Figure 2** d) Acetate (mg/l) vs. Time (h) for different initial conditions under aerobic conditions*

From a global point of view, the results obtained were quite satisfying. Indeed, less than 15% difference between the data and the model simulations are observed whatever the experimental and initial conditions (which appears quite reasonable when dealing with the modelling of biological systems). In addition, the limitation and inhibition phenomena are quite well visible (in particular see the limitation by nitrate in both Figures 2a and 2b and, depending on the curves, either the inhibition or the limitation phenomena by acetate in Figures 2b and 2d).

Conclusions and perspectives

A model of the particular behavior of the strain *Microvirgula aerodenitrificans* for both aerobic and anoxic conditions has been proposed. The structural analysis of the models as well as the results of its theoretical identifiability and sensitivity have been investigated. The experimental results and those issued from simulations were confronted and discussed. The unification of these two models within one is now under investigation in order to take into account the influence of other aeration conditions, that is comprised between the complete anoxic or complete aerobic conditions.

References

1. Abeliovich and A. Vonshak (1992) Anaerobic metabolism of *Nitrosomonas europaea*. Arch Microbiol 158: 267-270.
2. Armaing C., Modélisation de la vitesse spécifique de croissance de la bactérie *Microvirgula aerodenitrificans*, Ing. Thesis, CNAM, Sciences de l'Ingénieur, Toulouse, France, (1998), 148 pages (in french).
3. Bock E., Schmidt I., Stuven R. and Zart D., Nitrogen loss caused by denitrifying *Nitrosomonas* cells using ammonium or hydrogen as electron donors and nitrite as electron acceptor. Arch Microbiol 163(1), (1995), 16-20
4. Bouchez T., Patureau D., Dabert P., Juretschko S., Doré J., Delgenes J.P., Moletta R. and Wagner M., Reasons for failure of the bioaugmentation of a nitrifying reactor : Monitoring the fate of the added bacteria and the response of the microbial community, (1999), Submitted in *Environ. Microbiol.*
5. Jetten M.S.M., Strous M., Van de Pas-Schoonen K.T., Schalk J., Van Dongen U.G.J.M., Van de Graaf A.A., Logemann S., Muyzer G., Van Loosdrecht M.C.M., Kuenen J.G., The anaerobic oxidation of ammonium. FEMS Microbiol Reviews 22, (1999), 421-437.
6. Mulchandani R. and Bernier M.P., Batch kinetic of microbial polysaccharide biosynthesis, *Biotechnology and Bioengineering*, 32, (1988), 639-646.
7. Patureau D., Davison J., Bernet N. and Moletta R., Denitrification under various aeration conditions in *Comamonas* sp., strain Sgly2. FEMS Microbiol Ecol 14, 1, (1994), 71-78.
8. Patureau D., Etude cinétique et physiologique d'une bactérie dénitrifiante en conditions aérobie. Suivi en réacteur parfaitement mélangé en culture pure et en culture mixte associée à une flore complexe", PhD Thesis in Microbiology, INSA, Toulouse, France, (1995), 189 pages.
9. Patureau D., Bernet N., Moletta R., Study of the denitrifying enzymatic system of *Comamonas* sp., strain SGLY2, under various aeration conditions with a particular view on nitrate and nitrite reductases, *Current Microbiol.*, 32, 1, (1996), pages 25-32.
10. Patureau D., Bernet N. and R. Moletta R., Combined nitrification and denitrification in a single aerated reactor using the aerobic denitrifier *Comamonas sp.* strain SGLY2, *Water Research*, 31, 6, (1997), 1363-1370.
11. Robertson L.A., Cornelisse R., De Vos P., Hadioetomo R. and Kuenen J.G. Aerobic denitrification in various heterotrophic nitrifiers", *Antonie van Leeuwenhoek*, 56, (1989), 289-299.
12. Robertson L.A. and Kuenen J.G. Combined heterotrophic nitrification and aerobic denitrification in *Thiosphaera pantotropha* and other bacteria", *Antonie van Leeuwenhoek*, 57, (1990), 139-152.

EVALUATION OF A CONTROL STRATEGY FOR BIOLOGICAL P-REMOVAL

M. OOSTERHUIS[1]*, H. SPANJERS[2], N. HVALA[3]
[1]TNO-MEP, Laan van Westenenk 501, NL-7334 DT Apeldoorn, The Netherlands
[2]Department of Environmental Technology, Wageningen Agricultural University, Bomenweg 2, NL-6703 HD Wageningen, The Netherlands
[3]Department Computer Automation and Control, Josef Stephan Institute, Jamova 39, SI-1000 Ljubljana, Slovenia

Abstract

Many respirometry-based control strategies are proposed in literature, however only a few are applied. This is due to a lack of insight in the behaviour of the controlled process under practical conditions. It is expected that the evaluation of control strategies by simulations can provide useful information about the performance of the controlled process under different conditions. With this essential information the implementation of proposed control strategies can be supported. In this article a control strategy proposed in the literature is evaluated by simulations. The control strategy should be used for an SBR-reactor with P-removal and is based on respirometry. From respirometric data it is possible to detect the end of COD uptake in the anaerobic phase and the end of P-uptake, PHA-consumption and nitrification in the aerobic phase as a sharp drop in the respiration rate. A similar reactor was modelled using a commercial simulation platform. With different influent concentrations the possibilities for implementing the controller were investigated. From this investigation it could be concluded that the end of the anaerobic phase is hardly detectable. It could also be concluded that varying influent concentrations of COD, P and N make it difficult to detect the end of the aerobic phase. A modification of the simulation model is needed to enable rigorous evaluation of the control strategy.

Key words: respirometry, control, simulation, P-removal, SBR, limitation knee.

1. Introduction

To remove phosphorus from wastewater the biomass should be subjected to alternating anaerobic/aerobic conditions (1). In the anaerobic phase P-accumulating organisms (PAO's) store Poly-β-hydroxyalcanoates (PHA's) from soluble low molecular fatty acids (S_f) in the wastewater. For this process the PAO's obtain energy by releasing phosphate from internal stored poly-phosphate. During the aerobic phase the PAO's grow on PHA and take up phosphate to store poly-phosphates for anaerobic or anoxic maintenance. To obtain good P-removing conditions in an SBR the anaerobic phase should be long enough to store all the S_f as PHA. The aerobic phase should be long enough to remove all the P from the wastewater. As a consequence these two phases can be stopped at the end of the corresponding processes. To reach a good N-removal the aerobic phase should be stopped at the end of the nitrification. Control of the phase duration is preferable to fixed phase duration because both the treatment time and the energy costs for aeration will be minimised. Alternatively, if the phase duration is fixed a phase can be too short to complete certain reactions, resulting for example in S_f-leakage to the aerobic phase or incomplete P-removal at the end of the aerobic period (5).

It is shown that in an SBR-reactor, the end of COD-uptake, P-uptake PHA-consumption and nitrification are detectable from the respiration rate as a sharp drop or knee (5). (The respiration rate during the anaerobic phase is the potential respiration rate in case oxygen is not limiting, and it can be measured in a continuously re-aerated sample from the reactor). Larose proposed to control the duration of both the anaerobic and the aerobic phase by detecting the specific drops in the respiration rate. He also did simulation experiments to investigate the effect of the control strategy on the P-removing process behaviour. For this a dynamic model for an enriched bio-P culture, proposed by Smolders *et al.* (6) was used. However, in these simulation experiments the respiration rate was not modelled and only the aerobic phase was controlled by measuring P-concentration instead of respiration rate. A complete evaluation by simulations of this control strategy has not been carried out yet. The aim of this research was to evaluate the described control strategy by simulations with the General model (3).

* Author to whom all correspondence should be addressed

2. Methods

An SBR with a volume of 15 m^3 was modelled in GPS-X (Version 2.3, Hydromantis, Inc, Canada, 1997).As process model the General model, developed by Dold (3), was used. The General model was derived from a combination of the ASM1 model for non-PAO's (4) and the model for PAO's (9) and includes 28 processes. The double exponential settling velocity model was selected for the settling process (8). In the simulation experiments we tried to approach the conditions of the experiments of Larose (5) as close as possible. However, we extended the anaerobic period to obtain COD-limitation before the end of the anaerobic phase. The cycle time was then 12 hours for each cycle which consisted of 7 hours anaerobic (first three min were used for rapid filling), 3.5 hours aerobic, 0.5 hour anoxic and 1 hour settling from which the last 20 minutes were used for decanting and wasting. With this cycle time a steady state situation was reached in, approximately, 30 days. To show all the limitation knees in the respiration curve, the aerobic phase was prolonged to 7.5 hours. The total cycle time was then 16 hours. The results shown in this paper are from a 16-hour cycle which was started from steady state reached from a 12 hour-cycle time.

Table 1. Cycle times and phase duration in hours.

Phase duration	Experiments (Larose, 1998)	Simulations to reach "steady state"	Simulations shown
Anaerobic	3	7	7
Aerobic	3.5	3.5	7.5
Anoxic	0.5	0.5	0.5
Settling	1	1	1
Total cycle	**8**	**12**	**16**

In each cycle, one batch of 7.5 m^3 wastewater was treated. At the end of each cycle 0.06725 m^3 volume of settled sludge was wasted,. The following influent concentrations were used to reach a steady state situation: S_S (complex soluble substrate): 300 mg COD/litre, S_P (soluble phosphorus): 20 mg COD/litre, S_{NH} (ammonium nitrogen): 28 mg N/litre.

Table 2. Influent compositions used in the simulations.

	1	2	3
COD (mg/l)	400	400	100
NH4 (mgN/l)	28	28	28
PO4 (mgP/l)	20	20	20

Three different influent compositions were simulated (Table 2) The following variables were monitored in the simulation experiments: the potential respiration was monitored in the anaerobic and the aerobic phase. This deduced variable is calculated from the state concentrations in the reactor, except oxygen, which is assumed to be non-limiting. The concentrations of ammonium (S_{NH}), nitrate (S_{NO}), PHA, S_P, S_S, heterothrophic biomass (X_{BH}), autothrophic biomass (X_{BA}) and *PAO's* were monitored as well. To obtain similar results as Larose (5), we changed some kinetic parameters in the model: Monod constants for aerobic growth of PAO's (*kbt1* and *kbt2*) were both reduced to 0.009 in order to increase the degradation rate of PHA by the same organisms. The rate of the conversion of *Ss* to *Sf* (*kc*) was increased to 4. For all the other kinetic and stoichiometric parameters and wastewater concentrations, the default values of GPS-X were used.

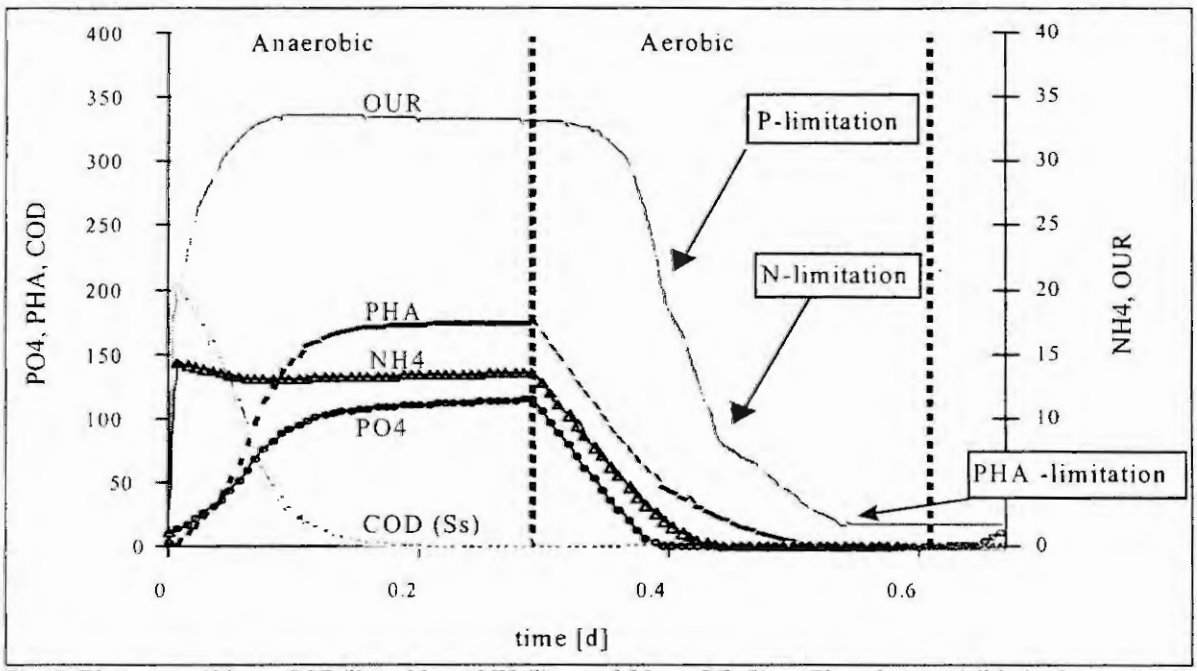


Fig. 1. First case: 400 mg COD/liter, 28 mg NH_4/liter and 20 mg PO_4/liter. Three knees visible in the respiration curve.

3. Results and Discussion

Simulations were carried out to detect the end of COD-uptake in the anaerobic phase and the end of P-uptake, PHA-consumption and nitrification in the aerobic phase. Three different influent compositions were investigated (Table 2).The results of the simulation of the first case are plotted in Fig. 1. It can be seen that the respiration rate increases in the beginning of the anaerobic phase. This is a consequence of the absence of the heterothrophic micro-organisms under steady state conditions (not shown). Because the remaining organisms (XBA's and PAO's) cannot grow directly on S_S and S_f the potential respiration rate on S_S and S_f is zero. The increasing rate in the beginning of the anaerobic period is due to the increasing concentration of PHA in the PAO's. In the experiments of Larose (5), a sharp drop was observed by the end of COD-uptake in the anaerobic phase suggesting that there were still heterothrophs in the SBR. When the model is used with default parameters the heterothrophs will still grow. In this case a limitation knee of COD in the respiration curve can be seen in the aerobic phase but the conversion of S_S to S_f and the consumption of PHA are unrealistic slow. A parameter fitting should be done to obtain realistic modelling results. The end of P-uptake, PHA-consumption and nitrification in the aerobic phase are clearly visible from the respiration curve (Fig 1.).

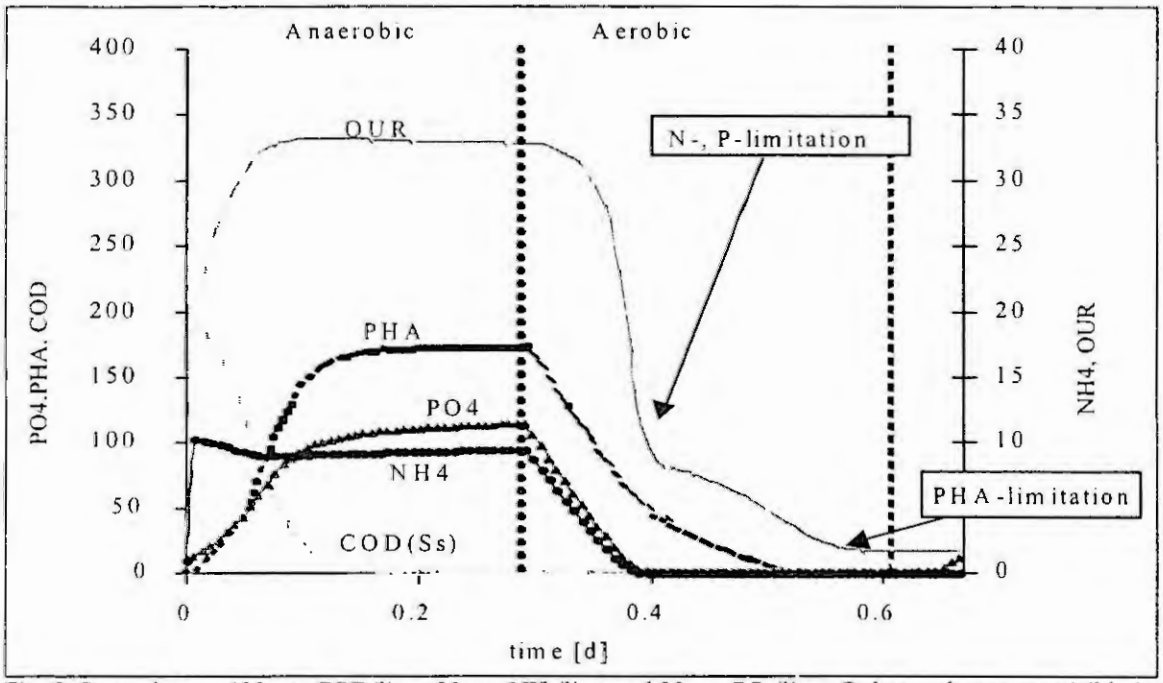


Fig. 2. Second case: 400 mg COD/liter, 20 mg NH_4/liter and 20 mg PO_4/liter. Only two knees are visible in the respiration curve.

In case 2 the NH_4-concentration in the influent was reduced to 20 mg N/l.. The simulation shows only two knees in the aerobic part of the respiration curve because both the NH_4-limitation and the P-limitation appear at the same time (Fig. 2). Such a situation complicates the identification of the separate knees. When it is unclear if a knee appears from NH_4-, P-, or PHA-limitation or from both P- and N-limitation, it will be difficult to design a good controller. It may, however, be possible to identify a certain limitation knee from its shape.

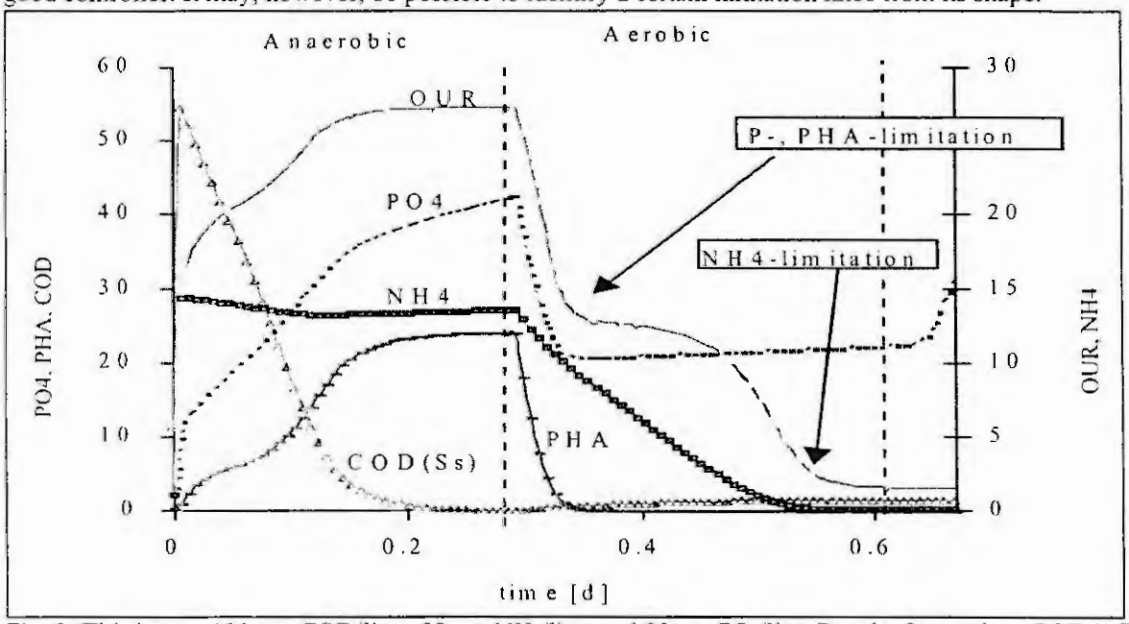


Fig. 3. Third case: 100 mg COD/litre, 28 mg NH_4/litre and 20 mg PO_4/litre Results from a low-COD influent.

In Fig. 3 the influent COD-concentration was reduced to 100 mg/litre. In this situation again only two knees are visible in the respiration rate. The first one appears from P and PHA and the second one appears from ammonium. In this situation the P- and PHA-knees appear in one knee because without PHA, P-uptake will not take place. Hence, two oxygen-consuming processes stop at the same time. Fig. 3 illustrates how the sequence of different limitations influences the number of knees. Additional output information from the process is needed to

define a control algorithm for the aerobic phase. In order to test the control strategy for the anaerobic phase, this model should be modified in order to allow the heterotrophic biomass to develop. For a good evaluation of a control strategy, a lot more has to be done. After the situation is modelled correctly, the controller should be implemented and tested under different circumstances.

4. Conclusions

From the simulations the following conclusions can be made:

- Evaluation of the aerobic phase duration control is possible with an adapted General model.
- Control of the aerobic phase is problematic because of two reasons:
  - the number of limitation knees are dependent on the wastewater composition
  - the limitation knees cannot be identified easily.
- For evaluation of the anaerobic phase duration control the General model has to be modified in such a way that a heterothrophic biomass will develop and the conversion of S_S to S_f is sufficiently fast.

5. References

1. Barker, P. S., Dold, P.L. Denitrification behaviour in biological excess phosphorus removal activated sludge systems. Water. Reearch, 30 (4), (1996), 769-780

2. Dold, P.L. Incorporation of biological excess removal in a general activated sludge model. In: Proc. 13th Int. Symposium on wastewater Treatment, Montreal, 1990, Canada, 83 – 113.

3. Dold, P.L., Ekama, G.A, Marais, G.v.R., A general model for the activated sludge process. Prog. Wat. Technol, 12, (1980), 47-77.

4. Henze, M., Grade, JR., C.P.L., Gujer, W. Marais, G.v.R and Matsuo, Activated Sludge Model no 1, Scientific and Technical Report, International Association on Water Pollution Research and Control (IAWPRC), IAWPRC. , London, 1987.

5. Larose, A. {Optimisation de la cyclologie d'un procédé de déphosphation biologique en reacteur biologique séquentiel par méthode respirométrique}, Optimisation of the cycle time of a biological P-removing process in a Sequencing Batch Reactor by respirometry, Département de Génie Chimique École polytechnique de Montréal. Montréal, Canada , 1998.

6. Smolders, G. J. F., Van Loosdrecht, M.C.M, Heijnen, J. J., A metabolic model for the biological phosphorus removal process, Water. Science & Technology,. 31 (2) (1995), 79-93

7. Spanjers, H., Vanrolleghem, P.A., Olsson, G., Dold, P.L. Respirometry in control of the activated sludge process: principles, IAWQ task group on respirometry, Wageningen, The Netherlands., 1998.

8. Takács, I., Patry, G.G. and Nolasco, D. A dynamic rnodel of clarification-Thickening process. Water Research,. 25 (10) (1991), 1263-1271.

9. Wentzel, M.C., Ekama, G.A., Loewenthal, R.E., Dold, P.L., Marais, G.v.R., Enhanced polyphosphate organism cultures in activated sludge systems. Part III: Kinetic model, Water SA, 15 (2) (1989), 89-102.

AUTOMATIC MODELLING OF CHEMICAL AND BIOLOGICAL SYSTEMS

David Schaich and Rudibert King
Measurement and Control Group
Institute of Process and Plant Technology, Sekr. P2-1
TU Berlin
Hardenbergstr. 36a, D-10623 Berlin, Germany

Abstract: Mathematical modelling and identification of physical systems where the structure of some equations is unknown, is a very difficult task and usually has to be done manually by a human expert. An experienced modeller will usually have a clear procedure to deduce a model. In a first step experimental data will be analysed on a qualitative level. Only those models are then tested in a quantitative identification which pass the qualitative check. In this contribution this efficient procedure is translated into methods and coded in a computer program. A tool *TAM-C* is introduced which automatically finds structures and parameters of formal kinetics of chemical and biological reaction systems. The main part of the contribution is dedicated to the qualitative preselection of possible model candidates. It will be demonstrated that the qualitative reasoning methods used in *TAM-C* play a crucial role for both an efficient and physically correct approach to the automated formulation of an accurate model.

Introduction

Mathematical modelling and simulation have become key techniques, central to all disciplines of science and engineering. In the field of chemical engineering, accurate models of dynamic processes are important for design, optimization, and model-based monitoring and control. Modelling a chemical process means the abstract description of the real physical system in a form of differential and algebraic equations, incorporating the concepts of mass and energy conservation, and the laws of thermodynamics. However, in many systems there are processes too complex or poorly understood to be purely modelled based on these principles. An example from the field of reaction modelling is the approximation of several elementary reaction steps by one formal reaction step, also called "lumping" in the chemical engineering literature, [4].

In these cases, appropriate mathematical structures for functions describing these processes in the differential algebraic equation system have to be found. This is usually done by a human expert in an iterative procedure using experience, "intuition", a priori and empirical knowledge of the system to be modelled and measurements. An experienced modeller will have a clear procedure to deduce a model. In a first step he or she will analyse experimental data on a qualitative level. Only those models are then tested in a quantitative identification which passed the qualitative check.

The focus of the work described here lies on the imitation of this human ability to reason about system structure and system behaviour on a qualitative level. The efficient qualitative methods developed are coded in a computer program together with well known quantitative procedures. A tool *TAM-C*, *T*ool for the *A*utomatic *M*odelling of *C*hemical reaction systems, is introduced which automatically finds structures and parameters of formal kinetics of chemical and biological reaction systems.

Qualitative Description of Measured Data

As already mentioned, an important characteristic of the model building process done by a human expert is that in a first step possible model candidates are proposed or rejected based on a qualitative analysis of the different experiments. Only promising model candidates which can at least explain all the observed qualitative behaviours of the system are further investigated.

The very same approach is used in *TAM-C*. In a first step a qualitative description of the measured data is automatically generated. To overcome measurement noise which is usually present in real data the time series are smoothed. Experimental and domain experience has to be used to choose the type of smoothing function. These curves are then divided into sections of the same qualitative state or behaviour.

In [3] a set of qualitative numbers is defined as the sign of the respective quantitative values, i.e. +,0 and -. Based on this, the qualitative state of a continuous quantitative variable $x(t)$ is defined as the

Type	A	B	C	D	E	F	G
x	⌒	⌒	⌒	⌒	/	\	—
dv	+	-	-	+	+	-	0
ddv	-	-	+	+	0	0	0

Figure 1: Episodes

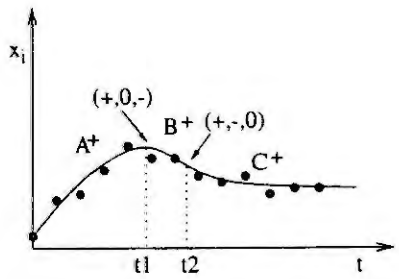


Figure 2: Abstracting noisy, quantitative measurement information (•) into a qualitative form by smoothing (–) and division into episodes ($A^+B^+C^+$).

triplet of the qualitative values of the variable and the first and the second derivative of the variable, [2]. A time interval, in which this triplet does not change, is called an episode. If any of the above properties changes value, a new episode starts. The time at which this change between two episodes occurs, is the transition time. The temporal shape of any smooth variable can described by such a sequence of episodes, called a history of episodes or qualitative history and associated transition times. All combinations of the signs for the first and the second derivative, which are physically possible, are shown in Fig. 1. Thus, all possible episodes are defined by the types from Fig. 1 and the sign of the value itself denoted by a superscript.

In Fig. 2, as an example, the smoothed curve reaches a maximum for the transition time t_1 (first dotted line), then decreases until it changes its curvature at t_2 (second dotted line) and reaches an equilibrium value, i.e. the qualitative history of the sketched curve in Fig. 2 is $A^+B^+C^+$. Thus, with this procedure, noisy, quantitative information can be transformed into a very simple qualitative representation, which contains the important qualitative features of the original time series.

Reasoning about system structure and behaviour

Any model proposed has to explain the observed behaviour at least qualitatively. Those models are rejected, which do not generate behaviours corresponding to the qualitative histories of measured variables. In an automated procedure a model generator in *TAM-C* proposes model candidates with different rate equations for the investigated reaction system, see Fig. 4.

Rule-based model library

A first check for any generated model candidate is done with a rule based model library. The rules compare properties of the model candidate with the observed qualitative behaviour. These rules comprise a priori and empirical domain knowledge as well as generated rules. The generated rules are results of qualitative simulations using an algebra based on episodes, see [8, 6]. With these simulations it can be deduced, which qualitative histories can be generated by which reaction system. The check with the rule based model library is very fast and efficient and most obviously unsuitable models are rejected at this early stage.

Dynamic qualitative identification

The remaining model candidates are then checked with a qualitative identification using dynamic interval simulations. Two methods are available for this: an order of magnitude identification, [5], based on a non-interacting interval simulation algorithm, and an interval identification, [7], based on an interacting interval simulation algorithm, which is described in more detail here. The difference between a non-interacting and an interacting interval simulator is that the later guarantees that no spurious, i.e. physically impossible, solutions are introduced, [1].

The qualitative interval identification method checks if the observed qualitative histories can be explained. This is illustrated with the following example. It is assumed that the model generator has created a possible model candidate to explain the data of Fig. 2. An interval simulation based on a first

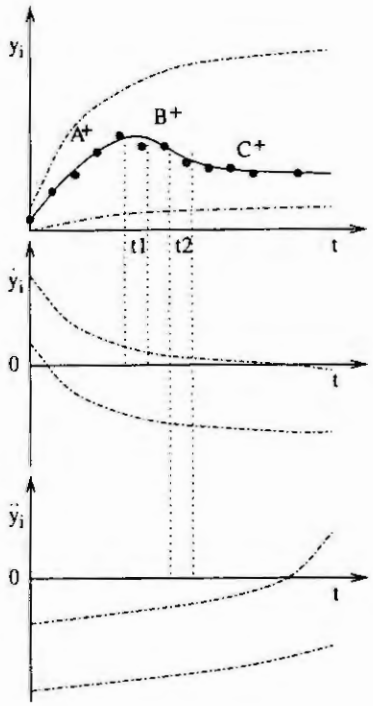


Figure 3: Sketch of the combined envelopes for y_i, $\dot{y}_i$, and $\ddot{y}_i$ $(- \cdot -)$ with the transition time intervals t_1 and t_2 $(\cdots)$ indicating the changes of the qualitative temporal shape of the smooth curve (–) from Fig. 2

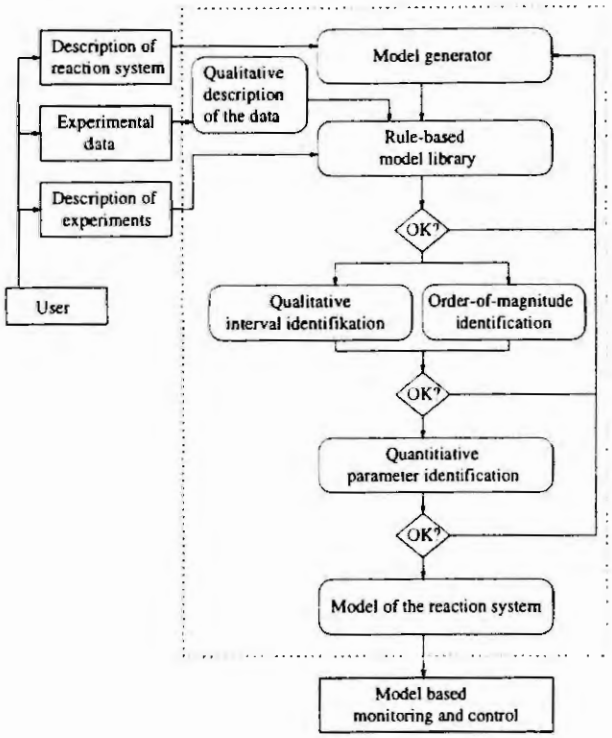


Figure 4: Modelling process in *TAM-C*

parameter guess (=intervals) now might yield a rather large band for y_i with which the model candidate cannot be excluded, see top of Fig. 3. To identify the qualitative behaviour of the system, the envelopes of the first and second derivatives are calculated as well: The first measured episode A^+ is consistent with the behaviour generated by interval simulation since the regions y_i, $\dot{y}_i$, and $\ddot{y}_i$ cover include the correct signs of this episode. The same is true for the change from episode A^+ to B^+, which corresponds to the crossing of the abscissa by the envelopes of $\dot{y}_i$ within an time interval around t_1. However, the transition from B^+ to C^+ cannot be explained, since the envelopes for $\ddot{y}_i$ cover only negative values for a time interval around t_2. Based on this, it can be concluded that the underlying process model which created the envelopes y_i, $\dot{y}_i$, and $\ddot{y}_i$ cannot explain the observed data points (•) in Fig. 3.

This example shows the main reason why the interval identification had to be included in *TAM-C*: On this basis it is very easy to check automatically whether an identification run can succeed. On a quantitative level it would be far more difficult to automatically rule out a nonlinear model. A "blind" search over the whole model space may yield a model which best fits the observations, but there is no guarantee that all important features of the system at study are captured.

Both methods also create good starting points for the subsequent quantitative identification. In the second stage of the identification process, the substantially more time-consuming quantitative structure and parameter identification, is applied only to the remaining candidates, which can at least qualitatively describe the measured data.

Modelling framework

All procedures depicted in the grey box in Fig. 4 are integrated in *TAM-C* and run without any interaction by the user. Therefore, the process of building adequate mathematical models of reaction systems is

substantially accelerated. Additionally, the modelling process is transparent and comprehensible for the user: Each method involved in the identification generates an explanation on why a certain candidate was accepted or not.

In addition to the methods described above, *TAM-C* has among other the following features:

- a model generator which is based on a symbolic mathematics defined in the class library *CLIMOS* [5]. It allows symbolic manipulations of all equations, e.g. analytic derivation, as well as qualitative computations with the same differential-algebraic equation system,
- a graphical user interface,
- and comprehensive library (*STaR*) for quantitative calculations (simulation, optimization).

Applications and Conclusion

The original goal of *TAM-C* was the automatic modelling of exothermic, safety critical chemical reactions. For most industrial exothermal batch- or semi-batch processes the thermal signal (temperature or heat evolution rate) is the only time series measurable in practice, because often all occurring compounds are generally difficult to analyse, highly toxic, or exist only for a very short time or under extreme conditions, respectively. This makes modelling of such processes generally very difficult. *TAM-C* has been used to successfully model several such reaction systems with several reactions steps, [8, 7, 5].

The application presented from this area is a process of AGFA Gevaert N.V., Belgium, to manufacture a speciality chemical. Recently, the scope has been extended to more complex reaction networks with special emphasis on biological systems, results for these systems will be given as well.

References

[1] A. Bonarini and G. Bontempi. A qualitative simulation approach for fuzzy dynamical models. *ACM Transactions on Modeling and Computer Simulation*, 4(4):285–313, 1994.

[2] J.T.-Y. Cheung and G. Stephanopoulos. Representation of process trends - part I. *Computers & Chemical Engineering*, 14:495–510, 1990.

[3] J. De Kleer and J. Brown. A qualitative physics based on confluences. *Artificial Intelligence*, 24:7–83, 1984.

[4] G.F. Froment and K.B. Bischoff. *Chemical Reactor Analysis and Design*. John Wiley & Sons, 2. edition, 1990.

[5] B. Münker. Entwicklung eines Software-Werkzeugs zur automatischen Modellierung chemischer Reaktionssysteme. Dissertation, TU Berlin, 2000.

[6] D. Schaich, S. Hellinger, B. Münker, and R. King. Automatische Erstellung mathematischer Modelle kritischer Reaktionssysteme und modellgestützter Fehlererkennungsverfahren. In UMSICHT Schriftenreihe Band 7, editor, *Rechneranwendungen in der Verfahrenstechnik, UMSICHT-Tage*, pages 3.1 –3.15. Fraunhofer Institut Umwelt-, Sicherheits-, Energietechnik, Fraunhofer IRB Verlag, 1998.

[7] D. Schaich, U. Keller, M. Chantler, and R. King. Interval identification - a modelling and design technique for dynamic systems. In *QR99, 13th International Workshop on Qualitative Reasoning*, Loch Awe, Scotland, 1999.

[8] D. Schaich and R. King. Qualitative modelling and simulation of chemical reaction systems. *Computers & Chemical Engineering*, 23(Suppl.):415–418, 1999.

A STRUCTURED MODEL FOR SIMULATION AND CONTROL OF FERMENTATION PROCESSES WITH COMPLEX MEDIUM COMPONENTS

Ch. Büdenbender and R. King
Measurement and Control Group
Institute of Process and Plant Technology, Sekr. P2-1
TU Berlin
Hardenbergstraße 36a, D-10623 Berlin, Germany

Abstract. Modern control of fermentation processes often includes model based control concepts where mathematical models are formulated to predict the behaviour of the process in question. Many secondary metabolites are produced under substrate limitation which usually cannot be described by unstructured models. It is shown, how the application of structured models can be supported by using a basic model structure valid for a variety of organisms. A structured model for growth of bacteria on defined media is used [1]. The model describes important mechanisms of bacterial metabolism and regulation and could already be applied to a number of different Streptomyces species [2]. Because of the modular character of the model special knowledge about the metabolism of the organism under consideration or the consumption of other medium components can be easily integrated into the basic structure. For identification of the parameters the model can be divided into subsystems which reduces the number of parameters to be identified simultaneously. The experimental data referred to in this contribution is obtained in fermentations with *Streptomyces griseus*.

Introduction

Many microorganisms produce secondary metabolites under substrate limitation. As the influence of a changing metabolism due to the shortage of substrates cannot be described by unstructured models the need for structured models is obvious. One of the main arguments against the use of structured models for process control, supervision and optimization is the expertise and time needed for the step of modelling and parameter identification. In [1] a model was presented for the description of growth and production behaviour of the bacterium *Streptomyces tendae* in defined media. The model was formulated following the principles of molecular genetics with the intention to obtain a basic model structure which can be used for a variety of organisms. In the present contribution the model is applied to fermentations with *Streptomyces griseus*. It is shown how the process of parameter identification can be supported by dividing the model into subsystems. The integration of additional knowledge about special metabolic phenomena is demonstrated for the example of the storage of phosphate and carbon which could be observed in the experiments with *S. griseus*. In a last step the application of the model to media containing amino acids is performed and an approach is presented for application of the model to complex media.

The modular concept of the structured model

The basic model structure as presented in [1] considers important mechanisms of primary metabolism and cell regulation which are common to at least a great number of bacterial species. The model divides the cell into compartments. For defined minimal medium the cell's inputs are the substrates ammonium, glucose and phosphate. Different groups of compartments can be distinguished. The precursors aminoacids and nucleotides are synthesized from the substrates. The macromolecules DNA, RNA and proteins are built from the precursors. The remaining cell material like lipids or fatty acids is comprised in the structural elements. The substrates and compartments as well as the culture volume are the state variables of the model and are balanced as follows:

- Cell compartments C_i: $\frac{d}{dt}m_{C_i}(t) = -\frac{Q_{out}(t)}{V(t)}m_{C_i}(t) + V_x(t)r_{C_i}(t)$
- Substrates S_i: $\frac{d}{dt}m_{S_i}(t) = Q_{in}(t)c_{in,S_i} - \frac{Q_{out}(t)}{V(t)}m_{S_i}(t) + V_x(t)r_{S_i}(t)$
- Culture volume $V(t)$: $\frac{d}{dt}V(t) = Q_{in}(t) - Q_{out}(t) + \sum_j Q_j(t)$

V_x denotes the cell volume. The reaction rates r_{C_i} and r_{S_i} are formulated using yield coefficients and kinetic expressions [1]. The dry cell mass is not a state variable as it can be calculated as the sum of the compartments. The consumption of glucose for maintenance is considered in the balance equation for glucose.

The division of the cell into compartments is illustrated in Figure 1. The compartments of the basic model structure as described above are shown in white. The flow of material from outside the cell into the compartments and between the compartments is illustrated by means of arrows. The processes of synthesis are marked with solid lines. Dashed lines are used to denote processes of decomposition. The dotted lines mark the consumption of glucose for energy requirements.

Identification of the model parameters

For identification of the model parameters a sequential quadratic programming algorithm is used. The substrates phosphate, ammonium and glucose, the macromolecules DNA, RNA and proteins as well as the dry cell mass are measured (marked with an asterisk in figures 1 and 2). For each of the measured quantities the sum of square errors between measurement and simulation is calculated.

The model as described above includes a set of 39 parameters. At first sight this seems to be a large number and the simultaneous identification of all parameters will not give satisfactory results. Therefore the model is divided into subsystems, that means only the parameters of the subsystem are made design variables while the other parameters are held constant. Figure 2 shows a possible subsystem for parameter identification. For this subsystem the sum of square errors between the simulated and measured values of ammonium, glucose and proteins is minimized. Like this the number of parameters to be identified simultaneously can be reduced and the performance of the optimizer is improved.

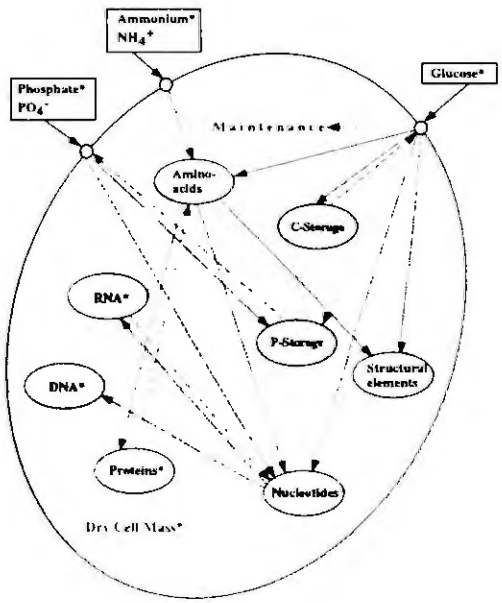


Figure 1: Division of cell into compartments

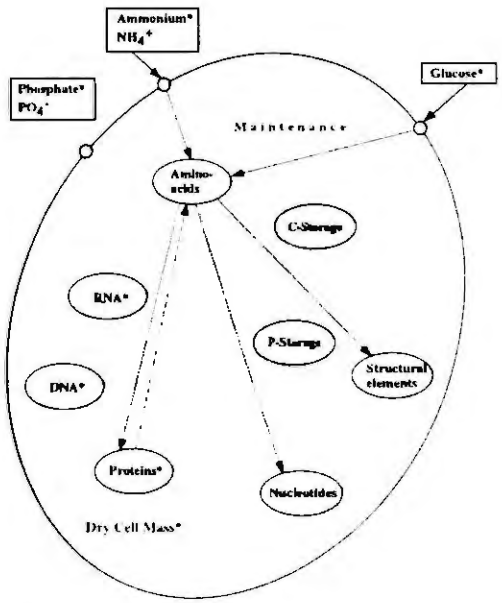


Figure 2: Division of model into subsystems

Integration of new cell compartments

With the basic model structure as presented in [1] a good fit of the experimental data could be achieved for *Streptomyces griseus*. The results are shown in figure 3.

Whereas the mechanisms considered in the basic model structure are common to at least a great number of bacterial species every organism is in a way specialized. This may be the production of secondary metabolites such as vitamins or antibiotics or the storage of substrates for periods of shortage. Evaluation of several fermentations with *Streptomyces griseus* suggested that the organism is able to store phosphate and carbon. Two independant dynamic storage compartments were integrated into the basic

model structure as illustrated in figure 2. With this modification a better fit of the experimental data could be achieved. Figure 3 shows a comparison of the simulation with the basic model structure and with the modified model for a fermentation with pulse addition of ammonium.

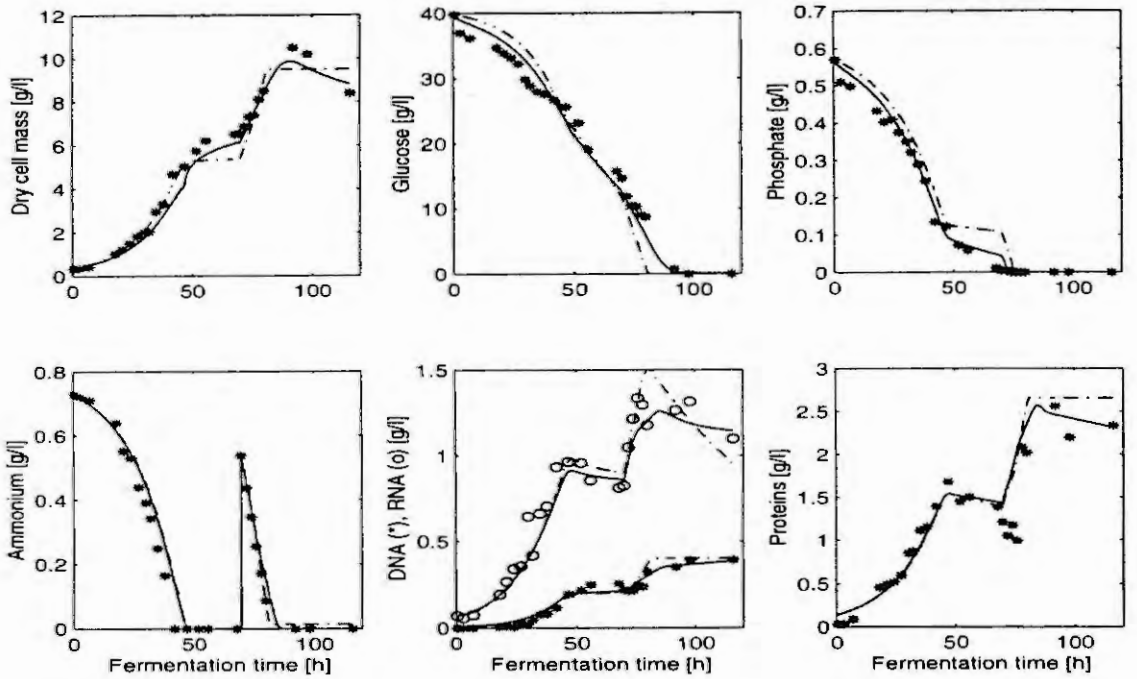


Figure 3: Simulation results with the basic model structure (dashed/dotted line) and the model including storage compartments for phosphate and glucose (solid line)

Integration of new medium components

An important argument against the use of structured models is that they are in general developed for growth in defined media. Whereas these media are superior in many respects and often used in the field of research it is common practice in industrial fermentations to use complex media as they are usually cheaper and easier to procure. If structured mathematical models shall be used for the control of industrial processes it is therefore undoubtedly necessary that they are able to describe growth and production behaviour in complex media.

Complex media usually contain large amounts of free aminoacids. The aminoacids are directly transported into the cell. There are different ways of aminoacid catabolism inside the cell. The most important is the desamination, where the aminogroups are stripped off and the carbon skeleton is metabolized. The approach chosen for model description is illustrated in figure 4. The external aminoacids are introduced as additional state variable. As for process control it is usually sufficient to know when all nitrogen from the medium is consumed the sum of all aminoacids in the medium is considered.

Some of the substrate aminoacids are directly transported into the internal aminoacid compartment. For the rest it is assumed, that all aminogroups are stripped off as ammonium which is added to the balance equation for ammonium. The carbon skeleton is regarded as glucose equivalent and is added to the balance equation for glucose.

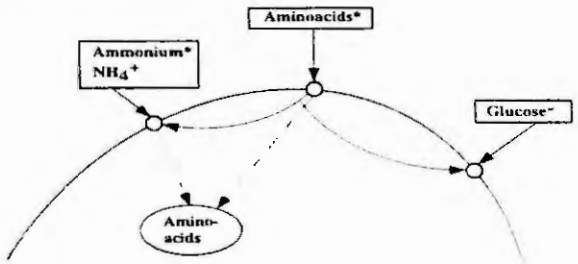


Figure 4: Amino acid consumption

The stoichiometric coefficients for the yield of ammonium and glucose by desamination depend on the composition of the aminoacids in the medium. For the fermentation shown in figure 5 where only Asparagine was added these coefficients can be calculated. For mixtures of aminoacids they have to be identified and may change during fermentation if not all aminoacids are consumed simultaneously. The advantage of this approach is that the formulation of growth on the substrates ammonium, glucose and phosphate can remain unchanged, so that the model can be used for defined media and media containing aminoacids. For fermentations with asparagine a good fit of the simulation to the experimental data could be achieved, as shown in figure 5.

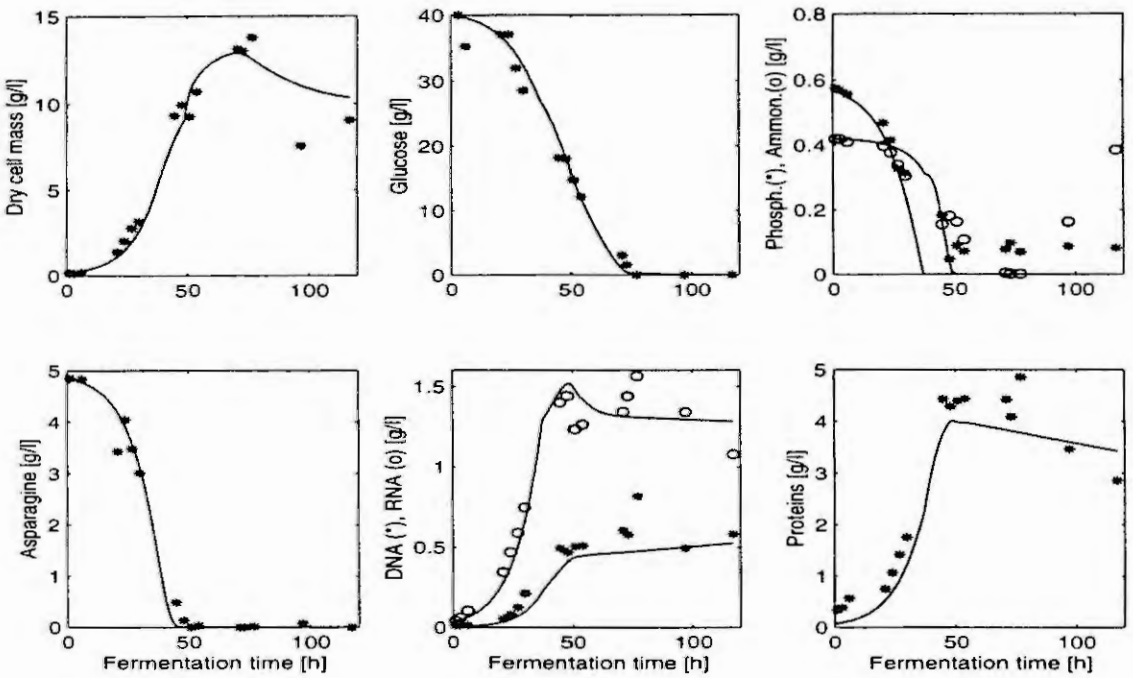


Figure 5: Simulation and experimental results for the addition of asparagine to the culture medium

Conclusions

In the present contribution it is shown how the application of structured models can be supported by using a basic model structure valid for a variety of bacterial species. This model could already be successfully applied for the description of growth of several Streptomyces species in defined media. Parameter identification can be improved by dividing the model into subsystems. Additional knowledge about the organism under consideration can be easily integrated as shown for the storage of phosphate and carbon. Also the consumption of further medium components can be integrated into the model without changing the basic model structure. With the approach presented in this contribution growth in definded media as well as in media containing aminoacids can be described with the same model. Like this the time and expertise needed for the step of modelling can be reduced drastically.

References

1. R. King, A Structured Mathematical Model for a Class of Organisms: 1. Development of a Model for Streptomyces tendae and Application of Model-Based Control. Journal of Biotechnology, 52 (1997),219-234

2. R. King, Ch. Büdenbender: A Structured Mathematical Model for a Class of Organisms: 2. Application of the model to other Strains. Journal of Biotechnology, 52 (1997),235-244

COMPUTER SIMULATION OF PROCESS GAS CIRCULATING SYSTEM

J . Shibata[1] , T . Mitani[1] and T.Matoba[2]
[1] Systems Design Core,Kanazawa Institute of Technology
7-1 Ohgigaoka Nonoichi Ishikawa 921 Japan
[2]Oita University
Dannoharu Oita 870 Japan

Abstract. This research is related to the computer simulation for the various kind of phenomena that happen simultaneously in the sinter bed layer. The mathematical model is constructed from heat exchange,reaction and mass transfer equations. These elemental equations are solved simultaneously through numerical calculation. The effect of process gas circulation can be caculated cleary in ideal case.

Introduction.

This report is concerned with the sintering machine of iron ore process illustrated in Fig.1. Recentry the exhaust gas circulation process reported[1] but the mathematical analysis la[1]lackes. In the sinter bed , many phenomena happen simultaneously. These process rapidly change since ignition time and moves down-ward of tndhe bed . To describe the sintering phenomena in details , the mathematical model is developed. In the conventional theoretical analysis, the coke combustion reactions have been taken into account so far but the decomposition reactions of limestone and also the permeability of the raw material bed have not been considered[2]. In this model the heat exchange between gas-solid ,the coke combustion reactions ,the vaporization of moisture,the re-condensation of vapor,the thermal decomposition of limestone ,the melting process of iron ore and the solidification process of molten ore are based previous model[3]. These are consisted of simultaneous non-linear partial diffenential equations including several unknowns. Performance of the sintering machine is affected by various factors, such as blend ratio of coke, the moisture content of raw material, blend ratio of limestone, melting properties of raw material, atmosphere conditions in ignition furnace . At first, the high temperature gas from the ignition furnace flows downward through the packed layer of raw materials ,then the heat exchange occurs between solid and gas.Then the volume of air passed through the sinter bed markedly affects the rate of heat transfer and the reaction of coke combustion in the packed bed. The moisture condensation zone forms at a high rate and reaches the bottom of the sinter bed in a short time. The gas flow resistance of the moisture condensation zone is larger than that of the initial raw sinter bed, so these phenomena affect each other. Therfore the raw materials are melt down and agglomerate each other.

Elemental equations.

The most fundamental phenomena in this process is the heat exchange between gas flow and solid. Here it can be discribed as the following equations from enthalpy balance.

$$\partial(V\cdot C\cdot T)/\partial X+\beta\cdot\alpha(T-t)=\partial(\varepsilon V\cdot C\cdot T)/\partial\Theta\cdots\cdots①$$

The enthalpy balance on solid side,

$$\beta\cdot\alpha(T-t)+Rc^{*}\cdot\Delta Hc+Re^{*}\cdot\Delta He=(1-\varepsilon)\partial(c\cdot t)/\partial\Theta+(1-\varepsilon)\rho_{s}\cdot\partial H/\partial\Theta\cdots\cdots②$$

The enthalpy change of melting,

$$\partial H/\partial\Theta=-(\partial H/\partial t)\cdot(\partial t/\partial\Theta)\cdots\cdots③$$

The function of melting material,

$$\int(dH/dt)\,dt=f\cdot(t_2-t_1)\,\Delta H_m\cdots④$$

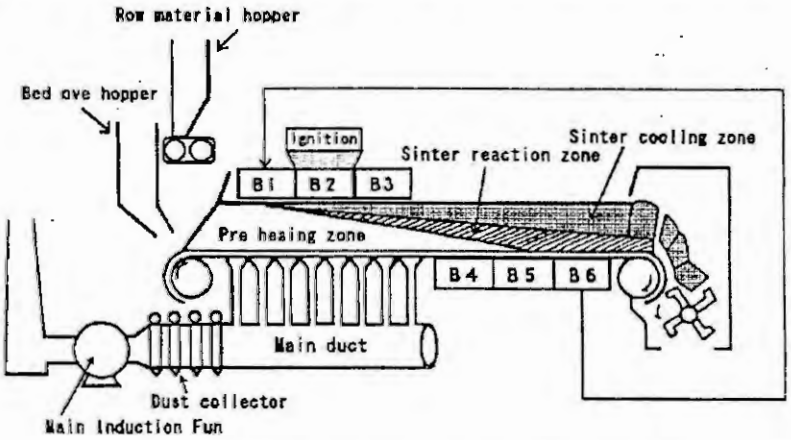


Fig.1 The model of the process gas rcirculating system

$$f = \Delta H_m / (t_2 - t_1) \quad \cdots\cdots ⑤$$

The coke combustion in sinter bed is the most important reaction. The main reaction of coke is the formation equation of the carbonic dioxide gas, it can be discribed as followings,

$$\gamma_c^* = 4\pi r^2 \cdot K_c^* \cdot Co_2 \quad \cdots\cdots\cdots ⑥$$

$$-\partial r / \partial \Theta = 4\pi r_e^2 (\rho_c / M_c) \cdot \gamma_c^* \quad \cdots\cdots\cdots ⑦$$

$$-\partial r_c / \partial \Theta = 4\pi r_e^2 (\rho_c / M_c) \cdot \gamma_e^* \quad \cdots\cdots\cdots ⑧$$

The reaction coefficient of carbon combustion ; K_r is expressed ,

$$Kr = K_c^* \cdot \sqrt{t} \cdot Co_2 \cdot \exp(-44000/R \cdot t) \quad \cdots\cdots\cdots$$

The overall reaction coefficient; K_c^* could be expressed,

$$K_c^* = 1 / Kf + 1 / Kr \quad \cdots\cdots\cdots$$

From carbon dioxide balance in the flow gas the next is discribed as following:

$$-\partial (V \cdot Co_2 / \rho) / \partial X + Rc^* + Re^* = \partial (\varepsilon \cdot Cco_2) / \partial \Theta \quad \cdots\cdots\cdots ⑨$$

The decomposition rate of lime stone,

$$\gamma_e^* = 4\pi r_e^2 \cdot k_e \cdot (Co_2^* - Co_2) \quad \cdots\cdots\cdots ⑩$$

Equation of the continuation on the fluid side,

$$-\partial V / \partial X + Mc \cdot Rc^* / \rho_c + Me \cdot Re^* / \rho_e = \partial (\varepsilon \cdot \rho) / \partial \Theta \quad \cdots\cdots\cdots ⑪$$

Material balance on oxygen gas can be discribed as

$$-\partial (V \cdot Co_2 / \rho) / \partial X - Rc^* = \partial (\varepsilon \cdot Co_2) / \partial \Theta \quad \cdots\cdots\cdots ⑫$$

In the case of developing the fundamental equations relevant to the drying process, the first stage of the drying process and the second stage of the falling rate period. Moisture condensation and drying process can be expressed from the material balance on the fluid. In wet zone,there are additional vapor condensation,then the ratio of the water in the solid raises gradually. Therefore,the condensation of moisture can be suppressed by reducing the humidity of the gas or raising the temperature of the raw sinter mix.

$$-\partial (V \cdot W) / \partial X = R_w^* \quad \cdots\cdots\cdots ⑬$$

From the material balance of the water content in the raw materials,

$$(1 - \varepsilon) \rho_s \cdot \partial w / \partial \Theta = R_w^* \quad \cdots\cdots\cdots ⑭$$

Enthalpy balance in the wet zone,

$$V \cdot C \cdot T) / \partial X + \beta \cdot \alpha (T - t) =$$

$$R_w^* \cdot \Delta H_w + (1 - \varepsilon) \partial (c \cdot t) / \partial \Theta \quad \cdots\cdots\cdots ⑮$$

Drying process in the constant rate period,

$$R_w^* = \beta \cdot \alpha \cdot (T - t) / \Delta H_w \quad \cdots\cdots\cdots ⑯$$

For the falling rate period is expressed,

$$R_w^* = \{\beta \cdot \alpha \cdot (T - t) / \Delta H_w\} \cdot (w - w_e) / (w_c - w_e) \quad ⑰$$

Through this drying process,the water in the upper solids move into the gas bulk flow as the vaper. The moisture condensation zone forms at a high rate and reaches the bottom of the layer in a short time. Then this vapor from the flow gas attach to the down of layer and are cooled by the lower part of solid, then the vapor condences again on the solid surface. Moisture condensation rate can be expressed as following,

$$R_w^* = V \cdot (\partial Ws / \partial T) \cdot (\partial T / \partial X) \quad ⑱$$

$$\partial Ws / \partial T = 0.005 \cdot \exp(0.057 \cdot T) \quad \cdots\cdots\cdots ⑲$$

Here Ws is the saturation humidity of suction gas. Gas flow resistance of the moisture condensation zone is larger than that of the initial raw bed. Pessure changes; P of each zone are the followings,

$$\partial P / \partial X = \kappa_1 \cdot \mu \cdot V + \kappa_2 \cdot \rho \cdot V^2 \quad ⑳$$

Table.1-a The Optimum of recycling system calculation (Input data)

Items \ Zones	Zone 1	Zone 2	Zone 3	Zone 4	Zone 5	Zone 6
Temperature (℃)	1413	20	20	20	20	20
Oxygen concentration (mol/m^3·10^{-3})	0.8	0.94	0.94	0.94	0.94	0.94
Humidity of gas (kgH$_2$O/kg air)	0.14	0.07	0.07	0.07	0.07	0.07
Gas Volume (kg air/m^3·min·10^{-3})	0.24	0.24	0.24	0.24	0.24	0.24

Circulation route	1.5	4.5	4	4	3	3

Table.1-b The Optimum of recycling system calculation (Result data)

Items \ Zones	Zone 1	Zone 2	Zone 3	Zone 4	Zone 5	Zone 6
Temperature (℃)	14	68	68	68	586	1137
Oxygen concentration (mol/m^3·10^{-3})	0.28	0.45	0.48	0.46	0.52	0.87
Humidity of gas (kgH$_2$O/kg air)	0.03	0.18	0.18	0.19	0.11	0.07
Gas Volume (kg air/m^3·min·10^{-3})	0.41	0.31	0.28	0.26	0.25	0.32
Average enthalpy (kal/kg ℃)	0.28	0.3	0.32	0.33	0.33	0.3
Layer section average temperature (℃)	261	437	533	617	599	408
Heat capacity (kcal air/m^3·min·10^{-3})	0.31	0.29	0.28	0.24	1.04	2.29

Here κ_1, κ_2 are pressure drop coefficients.

Caluculated result.

In general,partial diffrential equations are classified in three type. This mathematical model belongs to the second order hyperbolic type. This type partial differencial equation has two characteristic curve. On these curve,the equations of this model can be treated as the ordinary differential equations. It becomes more easier to be solved. The numerical solution are obtained on the integral curve. It was found from the computations that there existed a maximum step size of the difference as required for obtaining stable solutions and that the solutions could be expressed on the two curves regarding time and distance variables. On the basis of this model having manyoperation parameters,the numerical solutions can be obtained by computer based step by step calculation. Then the virtual operation can be achieved ,it was carried about eleven million steps for one operating case. Using the initial operating conditions only,this mathematical model which includes empirirical formulas of the pressure drop in each zone can predict the change in the suction air volume with the movement of moisture and the progress of sintering after ignition. As a result,it can be clarified that the temperature distribution in non steady state. The humidity concentration in the flow gas causes to decrease the partial pressure of oxygen concentration. The amount of moistere condensation largely depends on the difference between the wet bulb temperature of flow gas passing through the sinter bed and the temperature of the raw sinter mix. Using this model, the changes of the temperature distribution with the oxygen concentration were theoretically investigated. The critical concentration can be estimated as the changes of the temperature distribution of the bed layer with operating time. It is considered that the sintering reaction will not take place under this value. Thus,with the other operating conditions kept unchanged, the value is considered the lower limit of critical oxygen concentration. It can be also obsereved that there the same hesitation to rise solid temperature. It caused the coke reaction rate were not enough, the maximum temperature of solid can not be developed. While the concentration is raised over critical value,the temperature distribution expands along distance from the bed surface.This describes the mathematical model for sintering phenomena and analyzing clerly the inner situation of the sinter bed.

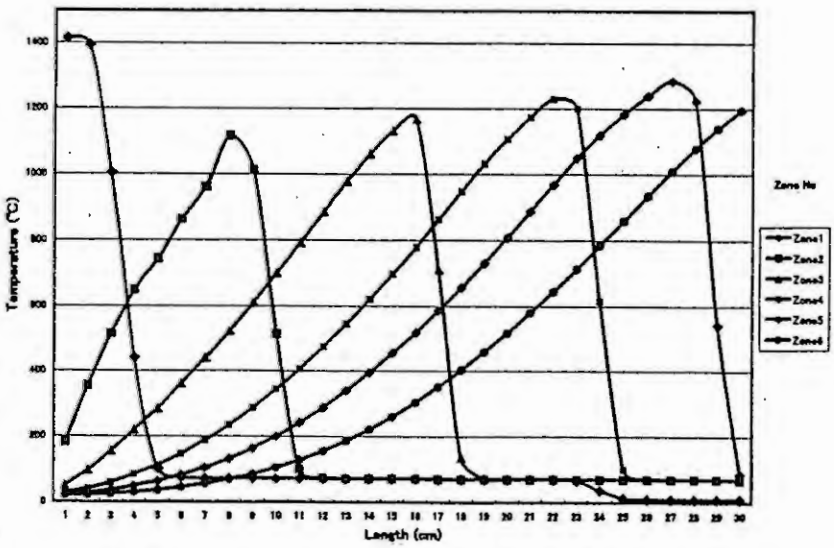


Fig. 2 Variation of temperature of transverse section of layer

Here caluculated three typical cases shown in Fig.2 and Fig.3.
The maximum temperature of solid tends to increase as the distance from the bed surface.

Summary.

The effects of operating factors on the reaction rates and the temperature distributions of the sinter bed can be quantitatively evaluated by simulation model.

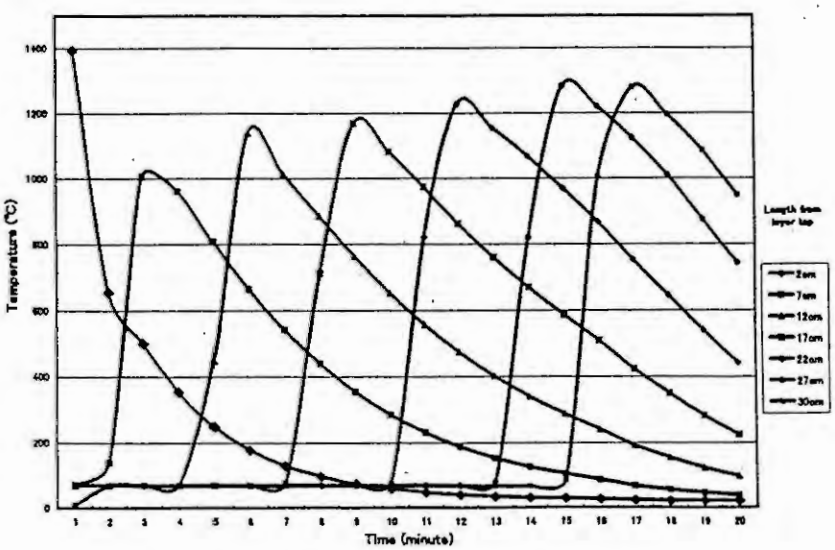


Fig. 3 Variation of temperature of layer in operation time

On the basis of the findings obtained by use of the model, there are some limitting condition to develop heat pattern in the bed. Here,it is clearly decided the critical concentration in the suction gas. The effects of these factors on the reaction rates, the temperature distribution of the sinter bed and critical conditions for the flue gas recirculation can be quantatively evaluated. It is seemed that high vapor existance decreases the coke combustion reaction. As a result, the waste gas recirculating method will increase the amount of moisture condensation in the sinter bed and decrease the maximum temperature in the bed.

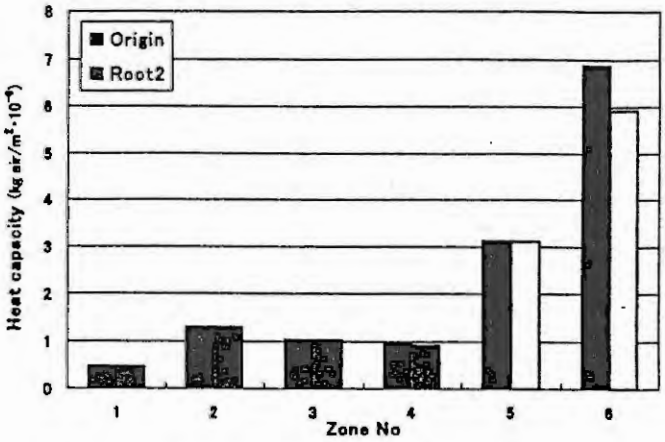


Fig. 4 Relative comperison of circulation effect

Then it should be paid enough compensation for the high temperature gas. By this virtual simulation technics developed here, the optimum condition can be dicided cleared for every cases. The results of these analysis have been utilized in developing the new tevhnics for the sintering process as well as improving the machine.

Nomenclature.

Cco_2 ,C : carbon dioxide concentration, specific heat of gas,
Co_2 : oxygen concentration , c : specific heat of solid
H :melting variable , Mc : molecular weight of carbon ,
Me : molecular weight of limestone ,
r : radius of coke particle , r_e :radius of limestone particle,
Rc*:coke reaction rate , Re*:limestone reaction rate
R_w*:moisture moving rate ,
T :gas temperature , t:solid temperature ,
V :suction gas volume ,
w_c :critical moisture content , W :moisture content of gas , Ws :saturated moisture content,
w_e :equilibrium moisture content , w :moisture content of solid ,
X :distance from bed surface ,
α :heat transfer coefficient , β :specific surface area,
γ_c*:coke reaction rate for particle , γ_e*:limestone reaction rate for particle,
ΔHc:heat of coke reaction, ΔHe:heat of lime stone reaction
ΔH_w :latent heat of vaporization,
ε :voidage of bed , Θ:time since ignition ,
ρ_c ;density of coke ,
ρ :density of gas , ρ_s ;density of bed

References.

1. J,.Rengersen E,Oosterhuis W,F,de Boer T,J,M,Veel J,Otto , First industrial experience with partial waste gas recirculation in a sinter plant. : La Revue de Metallurgie-CIT, Mars ,1995 , P.329-335
2. Z , Zhiyi , H, Tianzheng, Y,Xiaosheng , Mathematical model and computer simulation of moisture transfer process during sintering. : Transactions of NFsoc,Vo5, No.1,1995 ,p.15-20.
3. J.Shibata , Analysis of sitering process by mathematicall model. : Proceeding of the 6th International Conference on Mathematical Modelling, Vol.11,1988,p.956-961

ROBUST PREDICTIVE CONTROL AND NONLINEAR ESTIMATION FOR CHEMICAL REACTORS

O. Gehan, M. Farza, M. M'Saad

Laboratoire d'Automatique de Procédés L.A.P. EA 2611
I.S.M.R.A., Université de Caen, 6 Bd Mal Juin
F-14050 Caen cedex, France.
Fax : (33) 2 31 45 27 14 e-mail : msaad@greyc.ismra.fr

Abstract. A systematic approach is presented for the on-line control and monitoring of chemical processes. Predictive control and nonlinear estimation techniques are employed to achieve the required control performances while providing on-line estimates of the reaction rate. The main features of this approach are illustrated throw a typical chemical process.

1 Introduction

In the present paper, we present an approach for the on-line monitoring of chemical reactors. The control scheme of the proposed approach is based on a predictive control one. It makes use of a parametric Controlled AutoRegressive and Integrated Moving Average (CARIMA) model of the system [2]. While controlling the process, the proposed approach also provides on-line estimates of the reaction rates which constitute key parameters of the process. These estimates are issued from nonlinear observers which are designed on the basis of the process balance models. Two main features of the resulting monitoring approach are worth to be mentioned:

- The control design is carried out using the long range predictive control culture [2]. Of fundamental interest, the design parameter specification is carried using an iterative procedure that provides an appropriate shaping of the usual sensitivity functions, ensuring thereby the underlying robustness [5].
- The estimation procedure makes use of recent results on nonlinear observers' design [4]. The corresponding observers are synthesised on the basis of the process mathematical balance models. They do not assume or require any model for the reaction rates and are very successful in accurately estimating these variables.

We now propose to illustrate the main features of the proposed approach throw a typical chemical process. Indeed, the considered process deals with a chemical reaction with a single reactant A, held in a batch jacketed reaction calorimeter. The mathematical model of such a process is :

$$\begin{cases} \dot{T}_R = -\dfrac{\Delta H^T}{\rho_R c_{pR}} r + Q \\ \dot{C}_A = -r \\ \dot{T}_j = \dfrac{UA}{\rho_j c_{pj} V_j}(T_R - T_j) + \dfrac{F_j}{V_j}(T_{jin} - T_j) \end{cases} \tag{1}$$

where r is the reaction rate, C_A is the reactant concentration, ΔH^T is the reaction enthalpie, T_R (resp.T_j), ρ_R (resp. ρj) and c_{pR} (resp. c_{pj}) are respectively the temperature, the density and the specific heat of the reaction bulk (resp. the coolant/heating fluid), $F_j$is the coolant/heating fluid flow rate and T_{jin} is its inlet temperature. The heat flux Q is given by the following equation :

$$Q = \frac{UA(T_j - T_R)}{\rho_R c_{pR} V_R} \tag{2}$$

where U is the overall heat transfer coefficient, A is the heat transfer surface area and V_R is the reactor volume. Our control objective consists in tracking a desired profile for the reactor temperature. The plant input and output are T_{jin} and T_R respectively. While controlling the temperature, our aim also consists in obtaining an on-line estimate of the reaction rate r from the sole temperature measurements. Both control and estimation simulations have been carried out using SIMART package [6].

2 Control of the reactor temperature

For control purposes, we have identified a linear model of the system described by (1). Using a polynomial approach, the plant to be controlled was assumed to be approximated by a CARIMA model. Identification has been carried out around the usual setpoint $T_R(0)$ and in a least squares method sense using a pseudo-random binary sequence. One should notice that the heat production inside the reactor induced by the exothermic reaction is considered as an output perturbation.

A generalized predictive control law has then been synthetized on the basis of the obtained parametric control model. The remaining design parameters, namely the different horizons and the observer dynamic have been specified from an iterative procedure based on the shaping of the usual sensitivity functions. Such a procedure ensures a satisfying performances-robustness compromise [5] which is of prior importance to deal with the non-modelled dynamics such as the reaction heat production.

For comparison purposes, a PID controller has been synthesized using the relay method [1]. Figure 1 shows the Bode plots of the usual sensitivity functions corresponding to the robust predictive control and PID control systems. This figure clearly shows the great sensitivity of PID towards output noise measurements. This fact is confirmed by the temporal input behaviour. Indeed, Figures 2 and 3 show the input-output performances of both controllers submitted to two types of disturbances, namely noise output measurements and the exothermic reaction. Though both control systems provide acceptable controlled outputs, performances of the robust predictive controller are better according to the input behaviour.

3 Estimation of the reaction rate

While controlling the process output, on-line estimates of the reaction rate r have been provided by a non-linear observer which was synthetized on the basis of the mathematical balance model. The corresponding equations are [3] :

$$\begin{cases} \dot{\widehat{T}}_R = -\dfrac{\Delta H}{\rho_R c_{pR}} \hat{r} + Q - 2\theta(\widehat{T}_R - T_R) \\ \dot{\hat{r}} = \theta^2 \dfrac{\rho_R c_{pR}}{\Delta H} (\widehat{T}_R - T_R) \end{cases} \tag{3}$$

where $\theta > 0$ is the sole design parameter. In order to illustrate the performances of the proposed estimator, we have compared corresponding results with data issued from simulation, i.e. we have simulated the process model (1) by considering the following expression for the reaction rate :

$$r = k_0 exp(-\frac{E}{RT_R}) C_A \tag{4}$$

where k_0 is the reaction rate constant, E the activation energy and R the ideal gas constant. Estimation results are compared to data issued from simulations on figure 4 (corresponding curves are superimposed!). We remark the good performances of the observer in dealing with noise rejection and in tracking abrupt parameter variations.

4 Conclusion

An approach is presented for on-line control and observation of batch reactors. A generalized predictive control law is synthetized on the basis of a procedure which ensures an appropriate shaping of the usual sensivity functions. The control scheme is coupled with a nonlinear observer synthetized from the mathematical balance model of the process. The proposed approach provides an on-line estimate of the reaction rate while ensuring the control of the reactor temperature.

5 References

1. Åström, K.J.and Hagglund, T., PID Controllers : Theory, Design and Tuning, Instrument Society of America, 1995.
2. Clarke, D.W and Mohtadi, C., Properties of the Generalized Predictive Control, Automatica, 25 (1989), 859-875.
3. Farza, M., Busawon ,K. and Hammouri, H., Simple nonlinear observers for estimation of kinetic rates in bioreactors, Automatica, 34 (1998), 301-318.
4. Gauthier, J.P., Hammouri, H. and Othman, S., A simple observer for nonlinear systems - Application to bioreactors., IEEE Trans. Autom. Control, 37 (1992), 875-880.
5. M'Saad, M. and Chebassier, J., Commande prédictive des systèmes. Commande optimale, Diderot, Paris, 1996.
6. M'Saad, M. and Chebassier, J., SIMART : un Logiciel de l'Automatique et ses Applications, In : Proc. Journées des Logiciels d'Automatique, Nancy, France, 1997.

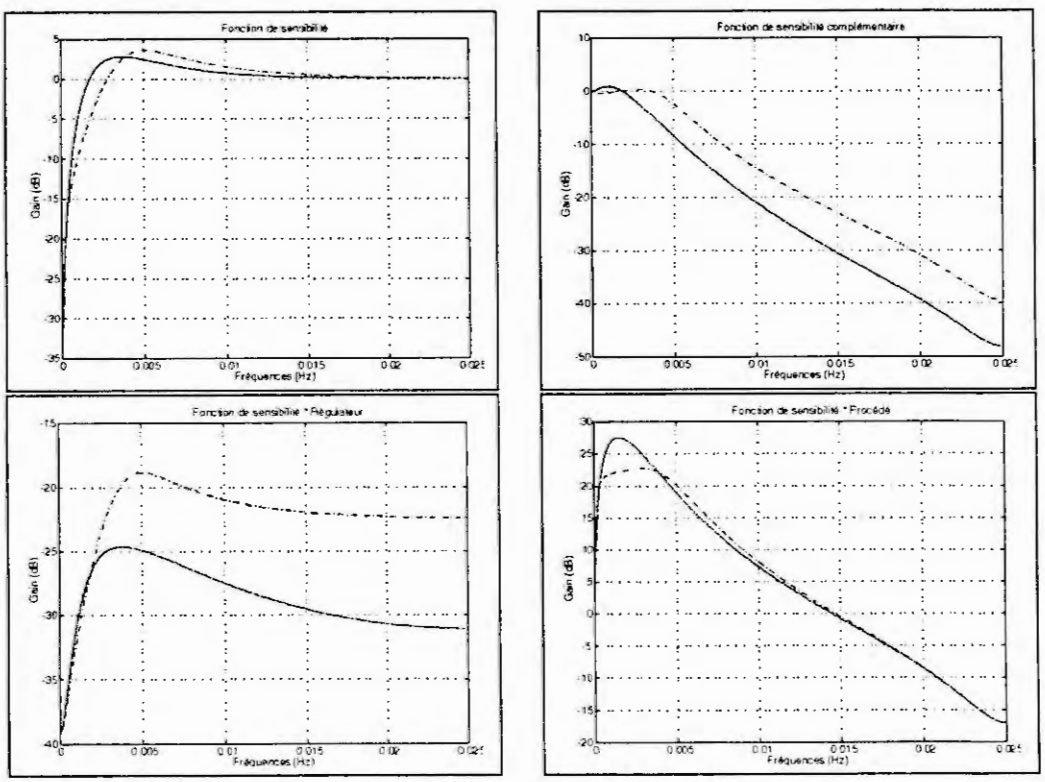


Figure 1: GPC and PID sensitivity functions ('–' : GPC; '-.' PID)

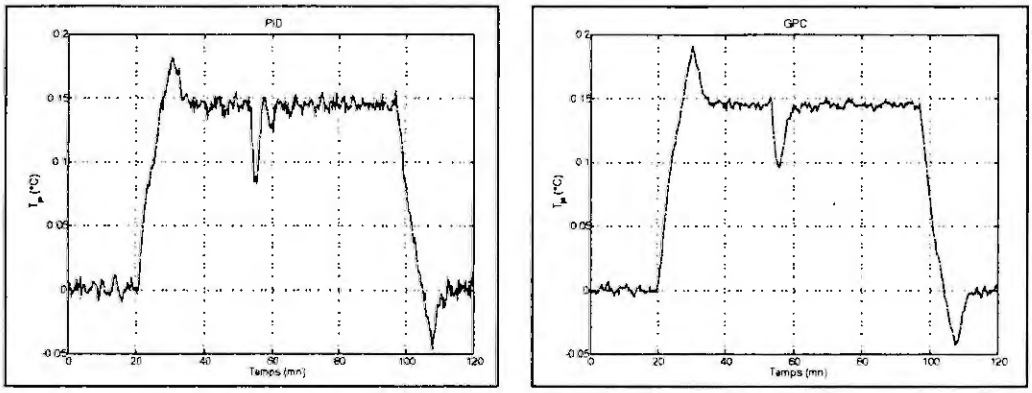


Figure 2: Time evolution of the jacket input temperature

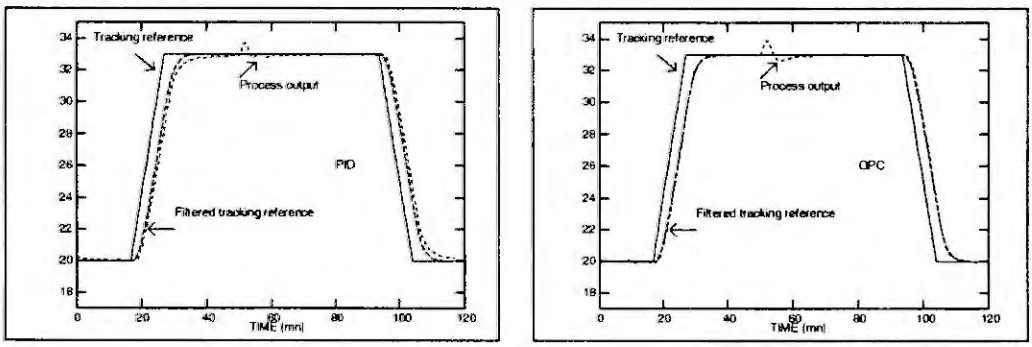


Figure 3: Controlled reactor temperature

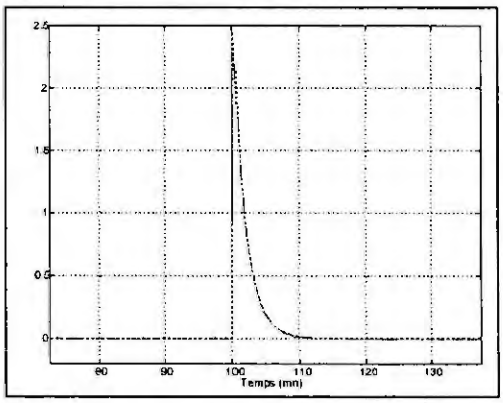


Figure 4: Comparison of the estimated and simulated reaction rate

ON THE IMPORTANCE OF TAKING SPACE INTO ACCOUNT WHEN MODELING MICROBIAL COMPETITION IN STRUCTURED FOODS

E.J. Dens and J.F. Van Impe
BioTeC - Bioprocess Technology and Control - Department of Food and Microbial Technology
Katholieke Universiteit Leuven - Kardinaal Mercierlaan 92, B-3001 Leuven (Belgium)
Fax.: +32-16-32.19.60; e-mail: jan.vanimpe@agr.kuleuven.ac.be

Abstract. Most of the models discussed up till now in predictive microbiology do *not* take into account the variability of microbial growth with respect to space. In *structured* (solid) foods, microbial growth can strongly depend on the position in the food and the assumption of *homogeneity* can thus *not* be accepted: *space* must be considered as an additional independent variable. In the current paper, a continuous time - two species competition model (proposed in previous work by the authors) is extended to take space into account. The spatio-temporal behavior of the spatially extended model is observed on a *coupled map lattice*. The smaller motility of the micro-organisms in solid foods allows spatial segregation which causes *pattern formation*. Evidence is given for the fact that taking space into account indeed has an influence on the behavior (coexistence/extinction) of the populations, which is very important in the field of predictive microbiology, where microbial safety is of major interest.

Introduction

As food safety is a growing concern in modern society, the scientific discipline of *predictive microbiology* gains worldwide more and more interest. The concept of predictive microbiology is that a detailed knowledge of the growth of micro-organisms in food products enables objective evaluation of the microbiological safety and quality of foods. If this microbial ecology is well understood, survival and/or growth of an organism of concern may be predicted on the basis of a mathematical relationship between microbial growth rate and environmental conditions. During the last decade, a large variety of predictive models for microbial growth have been developed. However, most of these models do *not* take into account the variability of microbial growth with respect to space. In *homogeneous* environments, like broths and fluid foods, this variability does not exist or may be neglected. In *structured* (heterogeneous) foods, microbial growth can strongly depend on the position in the food and the assumption of perfect mixing can thus not be accepted. In consequence, *space* must be considered.

In the field of ecology, [4] already pointed out that the spatially extended counterpart of an ecological model can lead to quite different behavior. The coexistence of two competitors is demonstrated although they have high interspecific competition coefficients. This coexistence implies the formation of structures over space with local segregation. A possible mechanism to control the level of pathogens in food products is to add an antagonist. Adding *Lactobacillus plantarum* could ,for example, be a way to inhibit the growth of *Escherichia coli* in a food. Obviously, when describing this system, it is of great importance to be able to accurately predict *coexistence* or *extinction* between these species. The knowledge that taking space into account has implications for population extinction urges researchers working in the field of predictive microbiology to consider this issue.

In this work, a continuous time - two species competition model (developed in a previous paper [2]) is extended to take *space* into account. The concerned model can be represented by a system of two non-autonomous differential equations:

$$\begin{aligned} \frac{dN_1}{dt} &= \frac{\mu_{max_1}}{N_{max_1}} \frac{Q_1(t)}{1+Q_1(t)} (N_{max_1} - N_1 - \alpha_{12} N_2) \cdot N_1 \\ \frac{dN_2}{dt} &= \frac{\mu_{max_2}}{N_{max_2}} \frac{Q_2(t)}{1+Q_2(t)} (N_{max_2} - N_2 - \alpha_{21} N_1) \cdot N_2 \end{aligned} \tag{1}$$

with

$$\begin{aligned} Q_1(t) &= Q_1(0) e^{\mu_{max_1} t} \\ Q_2(t) &= Q_2(0) e^{\mu_{max_2} t} \end{aligned}$$

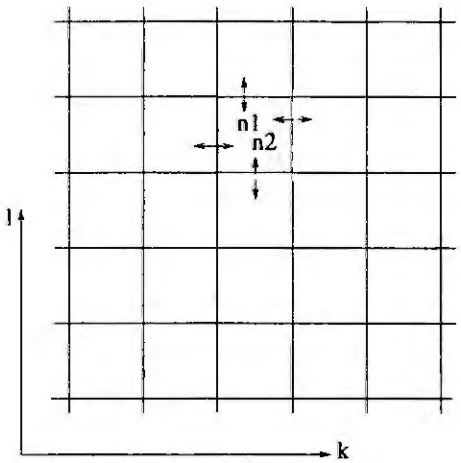


Figure 1: In order to take space into account, space is considered as a rectangular grid or lattice with k and l the spatial coordinates. Transfer of biomass from the current site (k,l) to the neighboring sites $((k,l-1),(k,l+1),(k-1,l),(k+1,l))$ –and vice versa– is possible.

In these equations, N_1 and N_2 [number of cells per unit of volume] are the cell densities of species 1 and 2 respectively. The variables Q_1 and Q_2 represent the internal physiological state of the cells of species 1 and 2 respectively. They grow according to an exponential law and allow the description of a lag phase for each of the two species. μ_{max_i} [1/time] indicates the maximum specific growth rate of species i ($i = 1, 2$), N_{max_i} [number of cells per unit of volume] represents the maximum population density of species i when no other species is present and α_{ij} [–] is a coefficient of interaction measuring the effects of species j on species i. In the cited paper, a stability analysis of the equilibrium points in function of these interaction coefficients is performed. However, the model as well as the results of the stability analysis are only valid for *homogeneous* food products since the model does not take into account the variability of microbial growth with respect to space. In *heterogeneous* foods, a different behavior is to be expected.

Introducing space

In order to take space into account, space is considered as a rectangular grid (or lattice), like presented in Figure 1. As such, in fact *two-dimensional* space is considered. The reason to adopt this two-dimensional configuration is in the first place because of the feasibility of experimental validation. Each lattice site is presumed homogeneous, such that the *cell densities* N_1 and N_2 at each separate lattice site grow according to model (1). At the separate sites, however, working with a *number of cells per lattice site* is more obvious. Therefore, the state variables n_1 and n_2 [number of cells/lattice site volume] are introduced at each separate lattice site. Transformation from the cell densities N_i to the number of cells per lattice site n_i depends on the *unit of volume* attributed to a single site. The state variables n_1 and n_2 still follow equations fo the same type:

$$\begin{aligned} \frac{dn_1}{dt} &= \mu_{max_1}\frac{Q_1(t)}{1+Q_1(t)}\frac{n_1}{n_{max_1}}(n_{max_1}-n_1-\alpha_{12}n_2) \equiv \mu_1 \cdot n_1 \\ \frac{dn_2}{dt} &= \mu_{max_2}\frac{Q_2(t)}{1+Q_2(t)}\frac{n_2}{n_{max_2}}(n_{max_2}-n_2-\alpha_{21}n_1) \equiv \mu_2 \cdot n_2 \end{aligned} \tag{2}$$

with n_{max_i} [number of cells/lattice site volume] the maximum number of cells of species i at a lattice site when no other species is present.

While at each separate lattice site, the number of cells of species 1 and 2 grow according to equations (2) transfer of biomass n_1 and n_2 from the current site to the neighboring sites –and vice versa– is possible (see also Figure 1). In situations of high cell density differences between the neighboring sites, this movement occurs in the direction of the site with the least dense concentration of cells. Indeed, a classical *balance model* for the accumulation of biomass at a certain site in a lattice is composed of two parts. On the one hand we have *biotransformation* (or evolution) of the cells at the lattice site in

$T = 10°C$		species 1: *E. coli*	species 2: *L. plantarum*
μ_{max_i}	[1/h]	0.05	0.06
N_{max_i}	[cells/ml]	$5.4 \cdot 10^9$	$2.4 \cdot 10^9$
$Q_i(0)$	[-]	0.0425	0.0425

Table 1: Numeric values for the maximum specific growth rate, the maximum population density [3][5] and the initial physiological state of the cells [1] for species 1 and species 2 at a temperature of $10°C$.

function of the environmental conditions at that particular site [equations (2)], and, on the other hand, there is *transport* of cells between neighboring sites:

$$\begin{aligned} \text{accumulation}(k,l) &= \text{biotransformation}(k,l) + \\ &\text{transport}((k,l),(k-1,l),(k+1,l),(k,l-1),(k,l+1)) \end{aligned}$$

Using an Euler approximation (discretization interval Δt) for the discrete solution of model (2), the extended model can be represented by the following set of difference equations:

$$\begin{aligned} n_1^{t+\Delta t}(k,l) &= n_1^t(k,l) + \Delta t\left[\mu_1^t(k,l)\cdot n_1^t(k,l)\right] + \Delta t D\nabla^2 n_1^t(k,l) \\ n_2^{t+\Delta t}(k,l) &= \underbrace{n_2^t(k,l)}_{\text{previous state}} + \underbrace{\Delta t\left[\mu_2^t(k,l)\cdot n_2^t(k,l)\right]}_{\text{biotransformation}} + \underbrace{\Delta t D\nabla^2 n_2^t(k,l)}_{\text{transport}} \end{aligned} \tag{3}$$

in which the transport of the cells of species 1 and 2 in one time step Δt from/to the current site (k,l) is defined as the diffusive operator ∇^2, which matches the following rule:

$$\begin{aligned} \nabla^2 n_1(k,l) &= n_1(k+1,l) + n_1(k-1,l) + n_1(k,l+1) + n_1(k,l-1) - 4n_1(k,l) \\ \nabla^2 n_2(k,l) &= n_2(k+1,l) + n_2(k-1,l) + n_2(k,l+1) + n_2(k,l-1) - 4n_2(k,l) \end{aligned} \tag{4}$$

The diffusion coefficient D [surface unit/time unit] is a measure for the firmness of the food. It matches infinity if the food is very fluid, and zero if no movement of micro-organisms is possible. In our application D is chosen to be very small, referring to a *solid* food.

Results

Consider, for example, a gel of 10 by 10 cm and 1 mm thick, in which micro-organisms grow. This surface is then superimposed with a grid of 100 times 100 sites, each site covering 1 mm^2 (and 1 mm thick). The unit volume attributed to a single site (*lattice site volume*) is thus 1 mm^3. Implementing the extended model equations (3) on this randomly initialized coupled map lattice (with *Neumann* boundary conditions), the spatio-temporal behavior of the extended two species competition model is observed. The model parameters necessary for model (2) are inspired by typical values for the pathogen *Escherichia coli* (N_1) in presence of the antagonist lactic acid bacterium *Lactobacillus plantarum* (N_2) (Table 1).

The steady state solution for a particular combination of interaction coefficients ($\alpha_{12} = 3.5$ and $\alpha_{21} = 0.6$) and a very low diffusion coefficient ($D = 16.7\mu m^2/s$, referring to a solid, structured environment) is shown in Figure 2. As one can observe, there is *pattern formation*, making *global coexistence* of species 1 and 2 possible. This pattern formation is caused by spatial segregation, due to a very low diffusion coefficient like is the case in *solid foods*. In [2] however, it was shown that for the same combination of interaction coefficients, *no coexistence* of species 1 and species 2 is possible in a *homogeneous medium*.

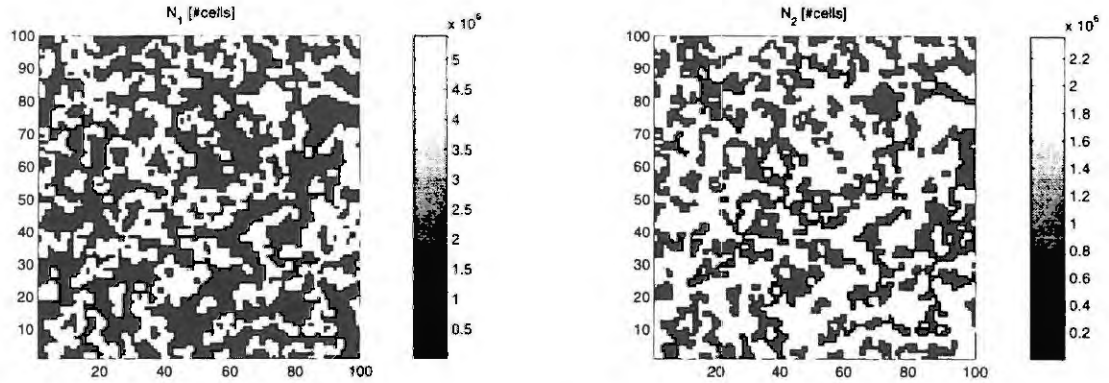


Figure 2: Steady state solution for microbial growth at a $100 \times 100\text{mm}^3$ lattice following equations (3) ($\alpha_{12} = 3.5$, $\alpha_{21} = 0.6$ and $D = 16.7\mu\text{m}^2/\text{s}$). The plots represent the density of species 1 (left) and species 2 (right) [#cells/lattice site volume] at each site of the lattice. The bar at the right maps the different shades of gray to its corresponding cell density.

This is because in fluid foods, no spatial segregation can persist. This is an example of fundamentally different behaviour in structured versus homogeneous foods.

Summary

In this paper, an illustrative example is given for the fact that microbial growth on a medium that is not perfectly mixed results in fundamentally different behavior than in homogeneous media, since spatial segregation causes pattern formation and has an influence on coexistence/extinction of populations. In the field of predictive microbiology, a correct prediction is crucial, e.g., when dealing with pathogenic micro-organisms in foods. The main message of this work is that, when modeling microbial growth in a structured environment, a more *extended model structure* which takes space into account is necessary for reliable predictions. This structure will certainly involve two parts: on the one hand *local microbial growth* in function of the local environment, and on the other hand *transfer of biomass* (and possibly other components, such as nutrients, metabolites, ...) between neighboring sites. More extensive review and exploration of such models applicable in the field of predictive microbiology will be reported in future work.

Author E. Dens is a research assistant with the Fund for Scientific Research-Flanders. Work is supported by the Research Council of the Katholieke Universiteit Leuven as part of projects COF/98/008 and OT/99/24, the Fund for Scientific Research - Flanders as part of project G.0267.99, and the European Union as part of project EU FAIR CT97-3129. The scientific responsibility is assumed by its authors.

References

[1] Baranyi, J. and Roberts, T.A., A dynamic approach to predicting bacterial growth in food. International Journal of Food Microbiology, 23 (1994), 277-294.

[2] Dens, E.J., Vereecken, K.M. and Van Impe, J.F., A prototype model structure for mixed microbial populations in food products. Journal of Theoretical Biology, (2000), (in press).

[3] Gill, C.O. and Phillips, D.M., The effect of media composition on the relationship between temperature and growth rate of Escherichia coli. Food Microbiology, 2 (1985), 285-290.

[4] Solé, R.V., Bascompte, J. and Valls, J., Stability and complexity of spatially extended two-species competition. Journal of Theoretical Biology, 159 (1992), 469-480.

[5] Zwietering, M.H., De Koos, J.T., Hasenack, B.E., De Wit, J.C. and Van 't Riet, K., Modeling of bacterial growth as a function of temperature. Applied Environmental Microbiology, 57 (1991), 1094-1101.

A MODEL OF SENSITIVITY AND CORRECTABILITY OF THE RECIPE COLOUR

B. Sluban
University of Maribor, Faculty of Mechanical Engineering, Smetanova 17, SI-2000 Maribor, Slovenia
E-mail: boris.sluban@uni-mb.si

Abstract. This paper presents a mathematical model of the colorant mixture colour sensitivity to the concentration errors. The items "recipe sensitivity" and "recipe correctability" are introduced as numerical quantities. The method of calculating the numerical estimates of the above quantities is developed. The results of a number of numerical experiments are used to illustrate the features of the theoretical model. In conclusion, a possible link between the predicted sensitivity values and the repeatability of the recipe colour is discussed.

1. Introduction

The prescribed target colour of a material can be achieved by the application of each one of a set of recipes (mixtures of various colorants in appropriate proportions). Not all recipes which match the prescribed target colour are equally appropriate for use in the production. Colourists prefer recipes which are the least sensitive to small random concentration errors and other inevitable small random variations in the coloration process. Another desired property is the sufficient recipe correctability (that means: small corrections of the recipe colour in any direction in colour "space" can be achieved by small adjustments of colorant concentrations). Colourists in industry select less sensitive (but still enough correctable) ones among various recipes using their knowledge gained from experience. In order to enable a quantitative comparison of the above two properties of various recipes in advance (i.e. already at the time of computer recipe formulation), a theoretical model of the sensitivity and correctability of the recipe colour is developed.

2. Theory

According to the needs of the coloration practise the items "sensitivity" and "correctability" of the recipe colour are introduced as numerical quantities. Let ΔE denote colour difference according to the formula CMC(l:c). The explanation of the colour difference formulae and the representation of colour by triplets of numbers – cylindrical coordinates L^* (lightness, the level on the vertical axis), C^* (chroma, the distance from the vertical axis) and h (hue, angle of rotation around the vertical axis) can be found e.g. in [1]. For the sake of simplicity, only the case of the recipe $\mathbf{c} = (c_1, c_2, c_3)$ consisting of the concentrations of three colorants will be considered. The application of the recipe $\mathbf{c} = (c_1, c_2, c_3)$ produces colour position $\mathbf{v} = (L^*, C^*, h)$. A small concentration change $\Delta\mathbf{c} = (\Delta c_1, \Delta c_2, \Delta c_3)$ produces a small colour change $\Delta\mathbf{v} = (\Delta L^* / (lS_L), \Delta C^* / (cS_C), \Delta H^* / S_H)$. Note that the colour difference ΔE is the length of the vector $\Delta\mathbf{v}$.

Let us introduce the directional sensitivity of the recipe $\mathbf{c}$ in the direction of a nonzero vector $\Delta\mathbf{c} = (\Delta c_1, \Delta c_2, \Delta c_3)$ (in concentration space) by:

$$s_{\Delta\mathbf{c}} = \lim_{t \to 0+} \frac{\Delta E(t\,\Delta\mathbf{c})}{\| t\,\Delta\mathbf{c} \|} = \lim_{t \to 0+} \frac{\| \Delta\mathbf{v}(t\,\Delta\mathbf{c}) \|}{\| t\,\Delta\mathbf{c} \|} \tag{1}$$

The sensitivities s_1, s_2, s_3 of the recipe $\mathbf{c} = (c_1, c_2, c_3)$ to the particular colorants are special cases of the directional sensitivity in directions $(\Delta c_1, 0, 0)$, $(0, \Delta c_2, 0)$ and $(0, 0, \Delta c_3)$, respectively.

The overall sensitivity of the recipe $\mathbf{c} = (c_1, c_2, c_3)$ is defined as the biggest directional sensitivity across all possible (nonzero) directions $\Delta\mathbf{c} = (\Delta c_1, \Delta c_2, \Delta c_3)$ of a move from the recipe position $\mathbf{c} = (c_1, c_2, c_3)$ in the concentration space:

$$s = \sup_{\Delta\mathbf{c} \neq 0} s_{\Delta\mathbf{c}} = \sup_{\Delta\mathbf{c} \neq 0} \left(\lim_{t \to 0+} \frac{\Delta E(t\,\Delta\mathbf{c})}{\| t\,\Delta\mathbf{c} \|} \right) \tag{2}$$

The correctability of a recipe is defined according to the concentration change $\Delta\mathbf{c}$ needed to produce the

colour change $\Delta\mathbf{v}$ requested. The directional correctability of the recipe $\mathbf{c}=(c_1,c_2,c_3)$ in the direction of a nonzero vector $\Delta\mathbf{v}=(\Delta L^*/(lS_L),\Delta C^*/(cS_C),\Delta H^*/S_H)$ (in colour space) is the number:

$$corr_{\Delta\mathbf{v}} = \left(\lim_{t\to 0+} \frac{\|\Delta\mathbf{c}(t\,\Delta\mathbf{v})\|}{\|t\,\Delta\mathbf{v}\|} \right)^{-1} \tag{3}$$

If the move in the direction $\Delta\mathbf{v}$ is impossible, we set $corr_{\Delta\mathbf{v}} = 0$. When opposite, note that $\Delta\mathbf{v}\neq\mathbf{0}$ implies $\Delta\mathbf{c}(\Delta\mathbf{v})\neq\mathbf{0}$. Special cases of directional correctability in the (nonzero) directions $(\Delta L^*/(lS_L),0,0)$, $(0,\Delta C^*/(cS_C),0)$, $(0,0,\Delta H^*/S_H)$, parallel to the L^* axis and to the chroma and hue lines, result in the correctabilities to the (CMC(l:c)-scaled) L^*, C^*, and h values. We do not present them in detail.

The overall correctability of a recipe is then defined as the lowest possible directional correctability across all possible directions $\Delta\mathbf{v}=(\Delta L^*/(lS_L),\Delta C^*/(cS_C),\Delta H^*/S_H)$ of the move from the actual colour position in colour space:

$$corr = \inf_{\Delta\mathbf{v}\neq\mathbf{0}} corr_{\Delta\mathbf{v}} = \inf_{\Delta\mathbf{v}\neq\mathbf{0}} \left(\lim_{t\to 0+} \frac{\|\Delta\mathbf{c}(t\,\Delta\mathbf{v})\|}{\|t\,\Delta\mathbf{v}\|} \right)^{-1} . \tag{4}$$

It is useful to note that for a corresponding pair of (nonzero) vectors $\Delta\mathbf{c}=(\Delta c_1,\Delta c_2,\Delta c_3)$ and $\Delta\mathbf{v}=(\Delta L^*/(lS_L),\Delta C^*/(cS_C),\Delta H^*/S_H)$, the following relation is valid:

$$s_{\Delta\mathbf{c}} = \lim_{t\to 0+} \frac{\|\Delta\mathbf{v}(t\,\Delta\mathbf{c})\|}{\|t\,\Delta\mathbf{c}\|} = \left(\lim_{t\to 0+} \frac{\|\Delta\mathbf{c}(t\,\Delta\mathbf{v})\|}{\|t\,\Delta\mathbf{v}\|} \right)^{-1} = corr_{\Delta\mathbf{v}} \tag{5}$$

Therefore, the value of the directional sensitivity in the direction $\Delta\mathbf{c}$ in concentration space is equal to the value of the directional correctability in the direction $\Delta\mathbf{v}$ in the colour space. Using the Eqn. (5), we see that the overall sensitivity of a recipe can also be interpreted as the biggest possible directional correctability:

$$s = \sup_{\Delta\mathbf{c}\neq\mathbf{0}} s_{\Delta\mathbf{c}} = \sup_{\Delta\mathbf{v}\neq\mathbf{0}} corr_{\Delta\mathbf{v}} \tag{6}$$

and that the overall correctability of a recipe can also be interpreted as the lowest possible directional sensitivity:

$$corr = \inf_{\Delta\mathbf{v}\neq\mathbf{0}} corr_{\Delta\mathbf{v}} = \inf_{\Delta\mathbf{c}\neq\mathbf{0}} s_{\Delta\mathbf{c}} \tag{7}$$

3. Numerical estimates

The Allen's iteration equation [2], [3] used in the recipe formulation algorithms has been transformed [4] in the form which links (small) concentration errors (changes $\Delta c_1,\Delta c_2,\Delta c_3$) with the visually relevant (small) changes $\Delta L^*/(lS_L)$, $\Delta C^*/(cS_C)$, $\Delta H^*/S_H$ of the perceived colour of the object. The resulting linear approximation formula:

$$(\Delta L^*/(lS_L),\Delta C^*/(cS_C),\Delta H^*/S_H)^{\mathrm{T}} = \mathbf{J}_{CMC}\mathbf{B}\,(\Delta c_1,\Delta c_2,\Delta c_3)^{\mathrm{T}} \tag{8}$$

is therefore approximately valid in a small volume around the recipe $\mathbf{c}=(c_1,c_2,c_3)$ in the concentration space and a small volume around the colour position $\mathbf{v}=(L^*,C^*,h)$ of the recipe in the colour space. This formula is then used to develop the numerical estimates of the above newly introduced quantities [4], [5]. Among other interesting results, it turns out that:

- the sensitivities s_1,s_2,s_3 of the recipe $\mathbf{c}=(c_1,c_2,c_3)$ to particular colorants are the lengths of particular columns of the matrix $\mathbf{J}_{CMC}\mathbf{B}$, and that
- the overall colour sensitivity and overall colour correctability of a recipe are the maximal and the minimal singular value of the matrix $\mathbf{J}_{CMC}\mathbf{B}$, respectively.

Usually, a larger number of recipes is treated. In order to reduce the amount of computation the following computationally simpler upper bound

$$\sqrt{s_1 + s_2 + s_3} \quad \left(= \|\mathbf{J}_{CMC}\mathbf{B}\|_F\right) \tag{9}$$

for the overall sensitivity can be used instead of the exact value. Also a lower bound for correctability can be developed and calculated.

4. Numerical experiments

A series of numerical experiments involving the calculation of recipe sensitivities and correctabilities (in a particular case of textile dyeing) for a larger set of target colours has been carried out.

The optical data of 8 basic dyes (2 yellows, 2 reds, 1 brown-red, 2 blues and 1 black) applied to textile fabric made of PAN fibres was used for recipe prediction. The target colours were chosen from the EUROCOLOR colour atlas, from 8 different L^*C^*-(half-)planes with hues 0, 125, 250, 350, 500, 650, 800 and 900 (per thousand), respectively. Targets have been spaced regularly by 10 units in L^* and C^* values.

For each target colour sensitivities to particular colorants, the upper bound (9) for overall sensitivity and the lower bound for correctability of all possible three-colorant recipes were calculated. The triplets consisting of the sensitvity upper bound (9), the correctability lower bound and the ratio (low. bound)/(upp. bound) for the recipes (containing the same combination of a yellow, a red and a blue colorant) for targets from the blue-green L^*C^*-plane of hue h = 500 are presented in Table I.

Table I. The triplets consisting of the predicted upper bound (9), lower bound (of overall correctability) and the quotient (lower bound / upper bound) of the recipes (containing the same combination of a yellow, a red and a blue colorant) for each of the blue-green targets indicated in L^*C^*-plane h=500 (EUROCOLOR 500.xx.xx).

	3404	2370	...sens. upp. bound	
L^*=80	190	152	...corr. low. bound	
	0.06	0.06	... low. b./ upp. b.	
	1884	1272	1139	
L^*=70	110	91	73	
	0.06	0.07	0.06	
	994	732	636	
L^*=60	63	55	44	
	0.06	0.08	0.07	
	507	402	372	391
L^*=50	37	32	26	21
	0.07	0.08	0.07	0.05
	272	214	199	235
L^*=40	21	18	14	12
	0.08	0.08	0.07	0.05
	141	112		
L^*=30	12	10		
	0.09	0.09		
Targets: h = 500	C^*=0	C^*=10	C^*=20	C^*=30

Table II. The predicted upper bounds (9) of overall sensitivities of the first 10 least metameric recipes for each of the blue-green targets indicated in the L^*C^*-plane h=500 (EUROCOLOR 500.xx.xx).The asterisk following some numbers in the table means that the average upper bound has been obtained from less than 10 values (at least from 4).

	4701	3109		... max.
L^*=80	3862	1740		... aver.
	2590	1010		... min.
	2660	1811	1516	
L^*=70	2001	1080	768	
	1211	580	404	
	1436	1020	849	
L^*=60	1117	621	433	
	645	352	229	
	747	610	497	525
L^*=50	575	350	252*	250*
	330	194	132	117
	406	334	267	317
L^*=40	311	190	134	150
	179	99	70	66
	215	181		
L^*=30	163	105		
	93	60		
Targets: h = 500	C^*=0	C^*=10	C^*=20	C^*=30

In addition, the upper bounds (9) for the sensitivity of the first 10 least metameric recipes per target were considered. These exhibited the same general trends as observed in the cases of recipes consisting of a single three-colorant combination treated above. To illustrate this, the triplets consisted of the maximal (above), average (in the middle) and minimal (beneath) of the mentioned 10 sensitivity upper bounds for targets in L^*C^*-plane with hue $h = 500$ are presented in Table II.

In the Tables I and II presented it can be seen that the lightest-shade recipes are the most sensitive ones (they generally have the biggest predicted upper bound of sensitivity) and that the recipe sensitivity rapidly decreases when the target gets darker. Furthermore, the predicted (upper bound of the) overall sensitivity is almost halved when we make a 10-unit decrease along each line parallel to the L^*-axis. The same trend was observed also for target colours in L^*C^*-planes for the 7 other hues considered. As higher lightness of a target in most cases (except e.g. for very saturated yellows) implies lower colorant concentration(s) in the recipe, the above observation is somewhat in accordance with the results of Alman's computer simulations [6].

The (upper bound of the) overall sensitivity of recipes varies less when the chroma C^* of the target is increased at the constant lightness level L^*. When the target is moved radially from the L^*-axis the recipe colour sensitivity moderately decreases in most directions, it can be almost halved at the border of gamut (see Table I) . This feature is in accordance with the observations that, generally, neutral shade recipes are more sensitive than others [7]. In our experiment but, in some of such radial directions (e.g. hue 650 of 1000) the recipe sensitivity can also moderately increase.

In Table II, it can be seen that for some targets the sensitivity bound of the most sensitive among the 10 recipes treated is up to 5-times higher than the sensitivity bound of the least sensitive one. The question arises whether such a distinct difference in predicted sensitivity to concentration errors also results in a significant difference in the repeatability of the (two) recipes considered. As concentration errors are only one of several sorts of small random errors that affect the repeatability [8], a simple and generally valid answer does not seem to exist. In situations where the relative contribution of concentration error to the total colour error is higher, such information can be useful. A series of experiments involving laboratory dyeing of textile fabric is being carried out with the aim to find a possible link between the predicted sensitivity to concentration errors and the repeatability of a recipe.

Summary

The items "sensitivity of the recipe colour to concentration errors" and "correctability of the recipe colour" are introduced as numerical quantities. Some interesting links and interdependence between sensitivity and correctability are noticed. The numerical estimates of the above quantities are presented. The results of a number of numerical experiments are used to illustrate the features of the theoretical model. A possible link between predicted sensitivity values and repeatability of the recipe colour is to be investigated in future work.

References

1. McDonald R., Colour Physics for Industry, 2nd Edition, Society of Dyers and Colourists, Bradford, 1997.
2. Allen E., Basic equations used in computer color matching, J. Opt. Soc. Am., 56, (1966) 1256-1259.
3. Allen E., Basic equations used in computer color matching, II. Tristimulus match, two-constant theory, J. Opt. Soc. Am., 64 (1974), 991-993.
4. Sluban, B. and Nobbs, J. H., The colour sensitivity of a colour matching recipe, Color Res. Appl., 20 (1995), 226-234.
5. Sluban, B. and Nobbs, J. H., Colour correctability of a colour matching recipe, Color Res. Appl., 22 (1997), 88-95.
6. Alman, D. H., Computer Simulation of the Error Sensitivity of Colorant Mixtures, Color Res. Appl., 11 (1986), 153-159.
7. Rieker, J. and Hilscher, K., Über die visuelle Relevanz von Farbstoff-Konzentration-Unterschieden auf gefärbten Textilien, Textil Praxis International, (1994), 499-501.
8. Sumner, H. H., Random errors in dyeing - the relative importance of dyehouse variables in the reproduction of dyeings, *J.S.D.C.*, 92, (1976), 84-99.

MODELING THE MECHANICAL ALLOYING PROCESS

Wolfgang Wiechert[1], Hippolyte Mournier[1], and Dietmar Hoppe[2]
[1]Abt. Simulationstechnik & Informatik and [2]Labor für Oberflächentechnik
FB 11, Universität Siegen, Paul-Bonatz-Str. 9-11, D-57068 Siegen
E-mail: wiechert@simtec.mb.uni-siegen.de

Abstract: Mechanical alloying is a process for the production of new materials. It takes place in a high energy mill where two powders of different origin are mixed. By the conversion of mechanical into chemical energy nanocrystalline structures emerge that cannot be produced by thermic alloying. The composition, physical properties and fine structure of the alloy can be influenced by various process parameters. However the chemical mechanism of the alloying process is currently not well understood. In order to study the influence of the process parameters on the product properties a general phenomenological model for mechanical alloying based on particle distributions is presented. It extends classical grinding models by an additional description of particle fusion. Some general considerations concerning the choice of breakage and fusion laws are undertaken and mathematical properties of the model like its transformation behavior and mass conservation are studied. Finally, simulation runs and a comparison with experimental data show that the alloying model is able to describe the main process characteristics.

Introduction

By mechanical alloying new materials like amorphous powders, intermetallic materials, solid solution alloys and metal matrix composites can be produced [5]. It takes place in a high energy shaker, planetary or ball mill where two powders of different origin are mixed (Figure 1). As an example an Al-Fe alloy with Fe particles of 10 nm diameter within an Al matrix can be produced which is not possible by classical thermic alloying [7]. The nanocrystalline fine structure of the product strongly depends on the physical and chemical properties of the raw materials. In particular the hardness of the materials plays an important role.

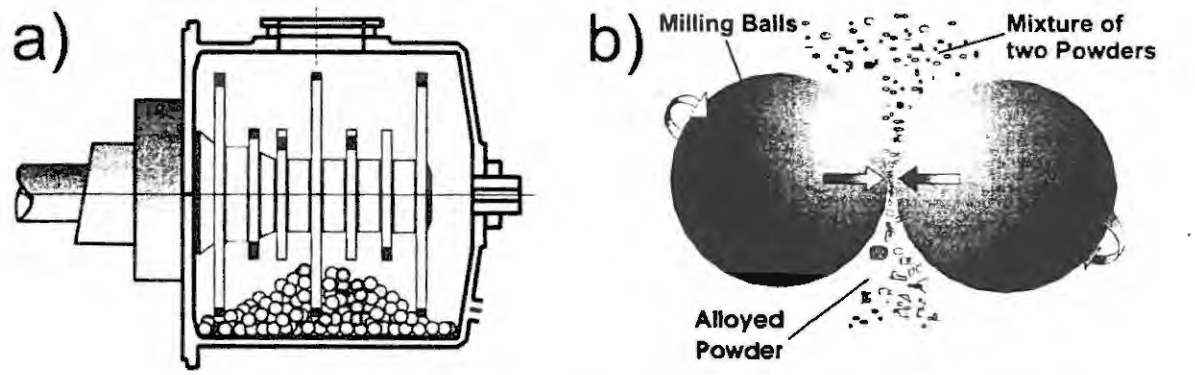


Abbildung 1: *a) Process of mechanical alloying in a high energy ball mill [4]. b) Nanocrystalline structures emerge by high energy impacts of particles.*

To understand the chemical mechanism of mechanical alloying the nanocrystalline structure of the alloy must be known at the atomar scale. The milled powder consists of particles that are composed from homogeneously structured crystallites (Figure 2b). These crystallites in turn belong to the different chemical substances which were present at process startup. The two different starting powders might already have a heterogeneous crystalline particle structure but the crystals are composed from the same chemical substance. Typical particle sizes of the milled powder are in the 10 μm scale while crystallites are in the range of 10 nm.

The space between the crystallites is filled by irregularly arranged "glueing" atoms from the two initially supplied sources (Figure 2b). These atoms connect the crystallites by chemical bonding. The chemical reaction mechanisms driven by the high energy particle collisions in the mill are currently not well understood. An established theory postulates that a high energy impact of two particles produces a local instable plasma that exists for only a few nsec [8]. In the plasma all atoms are deallocated from their original crystal position and establish temporary bonds to the other crystallite species. This state is quickly frozen by energy dissipation so that the thermodynamical equilibrium is not reached and a metastable intercrystalline phase remains.

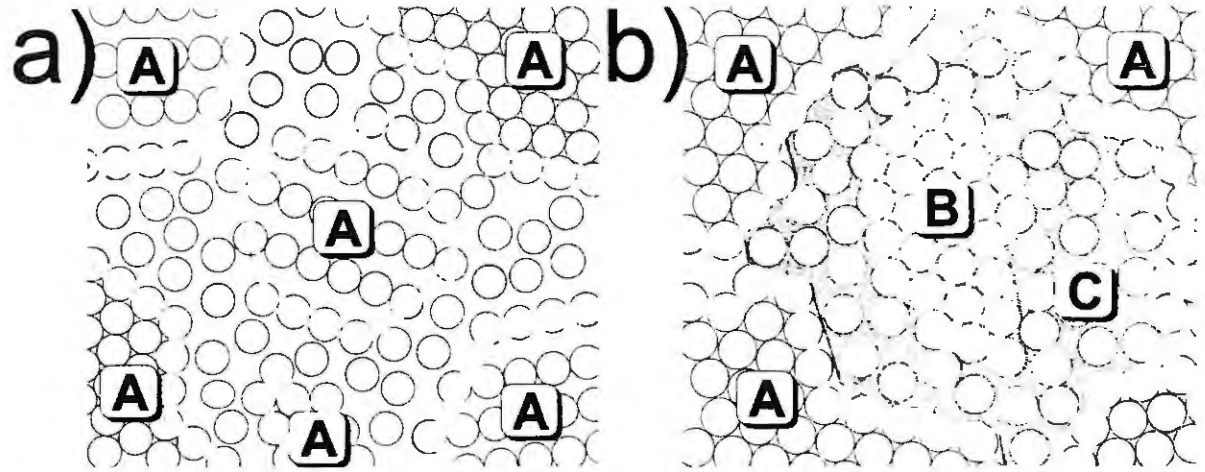


Abbildung 2: *Fine structure of a) the initially supplied powder and b) an alloyed particle. Nanocrystallites of type A and B are connected by intercrystallite bonding atoms C.*

Clearly, the properties of the alloy can be influenced by the initial amounts of materials, the initial particle size distribution, the milling time and the milling parameters. The investigation of these dependencies takes great experimental efforts [4]. For this reason mathematical modeling and simulation can be a helpful tool to obtain a deeper understanding of the process and to predict the product properties from the process parameters. However, no mathematical models for the mechanical alloying process are currently available. Consequently, this is a new demanding field for modeling and simulation.

Particle Distributions

In principle different modeling scales might be used to describe the mechanical alloying process. 1) A molecular dynamics approach on the atomar scale might help to understand the chemical mechanisms. 2) A crystallite interaction model on the mesoscopic scale can possibly describe the evolution of the particle fine structure. 3) A rough macroscopic particle distribution model can describe the overall composition of the particles. However because too few is currently known on the atomar scale and the mesoscopic scale requires to many modeling assumptions and parameters a first phenomenological approach is taken here at the macroscopic scale. Such a model can build on classical particle grinding models. On the other hand standard models for milling processes are not suited here because the most important event, the fusion of particles, is not taken into account.

Starting with classical grinding models and powders that consist of only one chemical phase the powder is described by a particle size distribution $u(x)$. The precise definition of $u(x)$ is obtained from the cumulative distribution function $U(x) = \int_0^x u(\xi)\, d\xi$ which is the total mass of all particles with a size smaller than x. Thus not the number of particles is counted for each particle size x but their total mass. This is a reasonable choice because the mass is conserved when particles break or fuse but not the number of particles. Particle size distributions are well established in classical grinding models. In particular, the available measurement instruments for particle distributions directly produce size distributions. However the concept of particle size is problematic because it depends on the definition of a size for any (not necessary spherical) particle shape.

As will become clear later the difference between size, volume or mass is not important in the classical milling models. However when particle fusion is taken into account the use of size distributions will lead to unnecessary complications. The reason is that volumes and masses are conserved when particles break or fuse but not the particle sizes. Assuming spherical particle shapes the transformation from size x (interpreted as the sphere diameter) to volume y is given by

$$y = f(x) = \pi/6 \cdot x^3 \qquad \Longleftrightarrow \qquad x = f^{-1}(y) = \sqrt[3]{6/\pi \cdot y} \quad . \tag{1}$$

The transformation from volume y to mass z is less problematic if the mass density ρ is known: $z = g(y) = \rho \cdot y \Longleftrightarrow y = g^{-1}(z) = z/\rho$. However, if the intercrystallite phase in an alloy is considered (Figure 2b), its mass density can only be estimated due to its irregular chemical structure.

Given the transformation (1) between size and volume the transformation between the corresponding particle size distribution $u(x)$ and the particle volume distribution $v(y)$ is given by the well known transformation formula for densities:

$$v(y) = \frac{u(f^{-1}(y))}{f'(f^{-1}(y))} = 2/\pi \cdot (6/\pi \cdot y)^{-2/3} \cdot u\left(\sqrt[3]{6/\pi \cdot y}\right) \tag{2}$$

Using this formula the classical models for the dynamics of particle size distributions can be transformed to volume distributions. Likewise the transformation to mass distributions $w(z)$ takes place which poses no problem because $g(y)$ is a linear function.

In the case of mechanical alloying a particle is composed from different chemical phases. If only two chemical substances A and B are considered and C denotes the intercrystallite phase (Figure 2b) the distribution $u(x)$ has to be extended to a three-dimensional distribution $u(a, b, c)$ where the vector $(a, b, c)^T$ specifies the composition of a particle from A, B and C. From this viewpoint tbe geometry of particles is neglected so that a particle composed from a few large crystallites may be characterized by the same vector $(a, b, c)^T$ as another particle with many small crystallites. For this reason the particle composition distribution approach is not completely satisfying because it cannot explain the particle fine structure.

Classical Grinding Models

The classical grinding model describes the time dependency of a particle size distribution $u(t, x)$ by an integro-differential equation [2]. Two phenomenological functions are required to describe particle splitting:

The **selection function** $s(x)$ specifies the probability per time unit that a particle of size x is selected for breakage. In reality this means that the particle is caught between two milling balls.

The **breakage function** $b(x, x')$ specifies the probability that a fragment of size x' is produced when a particle of size x breaks into parts. Clearly, it must hold $b(x, x') = 0$ if $x < x'$. Moreover the condition $\int_0^x b(x, x')\, dx' = 1$ is necessary for mass conservation, i.e. $b(x, x')$ is a probability density function for each fixed x.

Knowing s and b the grinding equation is obtained by mass balancing:

$$\dot{u}(x) = -s(x)\, u(x) + \int_x^\infty s(x')\, b(x', x)\, u(x')\, dx' \tag{3}$$

Clearly, this breakage model neglects the influence of particle shapes on the grinding process. Moreover the time dependency of s and b is usually not considered. It can be easily shown that this model is mass conserving. Many different parametric and nonparametric approaches to obtain reasonable selection and breakage functions can be found in the literature [9, 3]. For the sake of brevity the simplest approach assuming an exponential law for s and b is reproduced here:

$$s(x) = A \cdot x^\alpha \qquad\qquad b(x, x') = B \cdot (x'/x)^\beta \tag{4}$$

Here the breaking model assumes that only tbe relative size of x' compared to x determines the breakage probability. The parameter B cannot be freely chosen because it is already determined from the requirement that b is be a probability density for fixed x. Consequently, the complete system dynamics is governed by only three parameters A, α, β.

From the considerations given above it is interesting how the model transforms from size to volume distributions. Applying the transformation formulas it turns out that for the volume distribution $v(y)$ it holds

$$\dot{v}(y) = -\bar{s}(y)\, v(y) + \int_y^\infty \bar{s}(y')\, \bar{b}(y', y)\, v(y')\, dy'$$

with the transformed selection function $\bar{s}(y) = s(f^{-1}(y))$ and the similarly transformed breakage function $\bar{b}(y, y') = b(f^{-1}(y), f^{-1}(y'))$. This means that the transformed model has the same mathematical structure as the original model. Moreover if the exponential laws from Equation (4) are assumed the transformed selection and breakage models are also of the exponential type with constants: $\bar{A} = A \cdot (6/\pi)^{\alpha/3}$, $\bar{\alpha} = \alpha/3$ and $\bar{\beta} = \beta/3$.

Figure 3a presents an example simulation for the pure grinding process. The numerical implementation is based on a standard discretization of the particle distribution leading to a discrete particle class model [6]. If the integral in Equation (3) is replaced by some quadrature formula the mass conservation law need not hold for the

discretized equations. This has been enforced by an appropriate correction of the discretized breakage function. This discretization scheme is fully satisfactory for a first exploration of the parameter space. If the breakage and selection functions permit the splitting of arbitrary small particles the particle distribution will move towards smaller and smaller sizes with time and in the theoretical limit all the mass is concentrated at size 0.

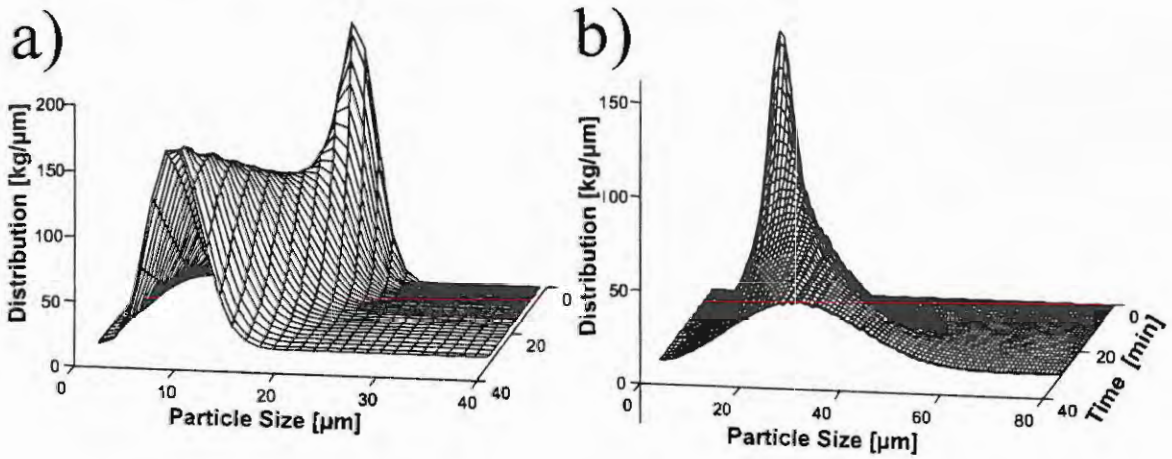


Abbildung 3: *Time course of a) a pure grinding process without particle fusion ($A = 1.25 \cdot 10^9/min, \alpha = 2, \beta = 4$) and b) a grinding-fusion process ($A = 1.25 \cdot 10^9/min, \alpha = 2, \beta = 4, C = 0.03$) with identical initial distribution.*

Grinding with Particle Fusion

The crucial step towards a general particle distribution model for mechanical alloying is the incorporation of particle fusion into the classical grinding model which has not been done before. Only fusion of two particles are considered because fusion of more than two particles are extremely rare and can be considered as a sequence of several fusion processes. As opposed to particle splitting where a probability density function b is required to describe the variety of possible fragment sizes, particle fusion is a purely deterministic event. Clearly, the target particle volume is determined as the volume sum of the fusion partners. This strongly suggests to use volumes or masses and not sizes to formulate the new equation.

A new law has to be introduced to describe the fusion probability. Particle fusion can happen when two particles are sufficiently close to each other and at the same time are situated between two colliding milling balls. If this geometrical constellation takes place the two particles fuse with a certain probability that depends on the collision energy of the balls. The probability that two particles of volume y and y' are sufficiently close in space is proportional to the product $v(y) \cdot v(y')$. The fusion probability is then expressed by the new symmetric welding function $w(y, y') = w(y', y)$ and the mass balance for the case of pure particle fusion is

$$\dot{v}(y) = -v(y) \cdot \int_0^\infty w(y, y') \cdot v(y')\, dy' + \int_0^y w(y', y - y') \cdot v(y - y') \cdot v(y')\, dy' \quad . \tag{5}$$

It can be proven that mass conservation is fulfilled. At the current state nothing precise is known about the dependency of the fusion probability $w(y, y')$ on the particle sizes. The simplest assumption is that the fusion probability of two particles trapped between colliding milling balls is independent of their size, i.e.

$$w(y, y') = \text{const} = C \quad .$$

To motivate this assumption it shall be interpreted in terms of particle numbers. Let there be n particles of size y and n' particles of size y' with a welding probability of $w(y, y')$. Assume for simplicity that $v(y/2) = v(y)$, i.e. the total volume of the $y/2$-sized particles equals the total volume of the y-sized particles. It follows that there are $2n$ particles of size $y/2$. Consequently, there are twice as many collisions of $y/2$-sized particles with y'-sized particles than there are collisions between y-sized particles with y'-sized particles. Assuming that the fusion probability is independent of the collision partner sizes the total volume of $y/2$-sized particles consumed by fusion with y'-sized particles is exactly the same as for the y-sized particles, i.e. $w(y, y') = w(y/2, y') = \text{const}$.

Obviously, the easiest way to formulate the fusion model is by using volume or mass distributions. In fact it turns out that Equation (5) does not behave well under the transformation to particle size distributions, i.e.:

$$\dot{u}(x) = -u(x) \cdot \int_0^\infty \overline{w}(x,x') \cdot u(x')\,dx' + \int_0^x F(x,x') \cdot \overline{w}(x', \sqrt[3]{x^3 - x'^3}) \cdot u(\sqrt[3]{x^3 - x'^3}) \cdot u(x')\,dx' \quad .$$

Although the structure of the first integral from Equation (5) is conserved with the transformed welding function $\overline{w}(x,x') = w(f^{-1}(x), f^{-1}(x'))$, the second integral changes and requires a correction term $F(x,x')$. In the simplest case of constant welding $F(x,x') = (1-(x'/x)^3)^{-2/3}$. Consequently, grinding-fusion models should always be formulated with volume or mass distributions. Nevertheless Figures 3 and 4 are rescaled to size distributions by using Equation (2) for better comparison with published results.

Figure 3b shows a simulation for a mixed grinding-fusion process according to Equations (3,5). The grinding parameters are exactly the same as in Figure 3a and a constant welding function is used. As can be seen the particle distribution now tends to a stationary limit distribution in which the grinding and fusion processes are in equilibrium. The asymptotic distribution does not depend on the initial distribution if the same total mass is taken. The model does not behave linear with respect to the starting mass because welding increases quadratically with the number of particles. In general if the fusion probability is increased the distribution mean moves towards higher values.

A milling experiment with 40 g Ag_3Sn powder was carried out to obtain a realistic particle distribution from a grinding-fusion process. A stable limiting distribution was observed [6] and by parameter fitting it could be shown that the simple model with exponential s and b functions and a constant w function can reproduce the experimental output. Moreover the corresponding parameters were determined with significant sensitivities. i.e. with good statistical quality.

A simple Model for Mechanical Alloying

A general model for the mechanical alloying process based on particle distributions must extend the concepts for splitting and fusion of particles presented above to a multidimensional mass distribution $v(t,a,b,c)$ where a, b and c denote the volume of the two crystallite phases A and B and the the inter-crystallite phase C (see Figure 2). The concepts of the selection, breakage and welding functions must then be extended to the multivariate case. Because the sum $a+b+c$ is the particle volume y the concept of selection, breakage and welding functions can be reused with y replaced by $a+b+c$. The only difference is that the target of a particle fusion between an $(a,b,c)^T$-sized particle and an $(a',b',c')^T$-sized particle does not necessarily have the composition $(a+a', b+b', c+c')^T$ because this would mean that no alloying takes place. Thus the details of mechanical alloying must be specified by a suitably defined generalized welding function $w(a,b,c,a',b',c')$.

Although this will lead to a general model in a rather straightforward way this approach will not be followed up in the present contribution because the computational effort is quite high. Instead a much simpler model is favored here by taking an idea from [1]. It might be considered as a very coarse discretization of the $(a,b,c)^T$ space. All particles are collected into three groups. The first group is characterized by at least 90 % A crystallites, the second by at least 90 % B crystallites and the third group C collects all other particles. Corresponding to these three groups there are three volume distributions $v_A(y), v_B(y), v_C(y)$. Breakage takes place within each of the groups separately so that there is one selection and breakage function $s_i(y), b_i(y), i = A, B, C$ for each group. Welding is more difficult because the group membership of a fused particle does not only depend on the group of the collision partners but also on their size. The following fusions are possible:

A + A	→	A	A + B	→	A	or	B	or C
B + B	→	B	A + C	→	A	or	C	
C + C	→	C	B + C	→	B	or	C	

For each of these cases a welding function $w_{i+j\to k}$ is defined. If multiple targets are possible the corresponding welding functions are appropriately weighted. The resulting lengthy model equations are not reproduced here. Figure 4 presents the outcome of a simulation run where the A and B powders have been supplied with the same initial distribution. The distribution of the alloy C finally becomes stable and A,B are completely used up.

Conclusions

Clearly, the three species particle distribution model is only a rough approximation to a general multidimensional distribution model. However it qualitatively reproduces the basic features of the alloying process. Based on the

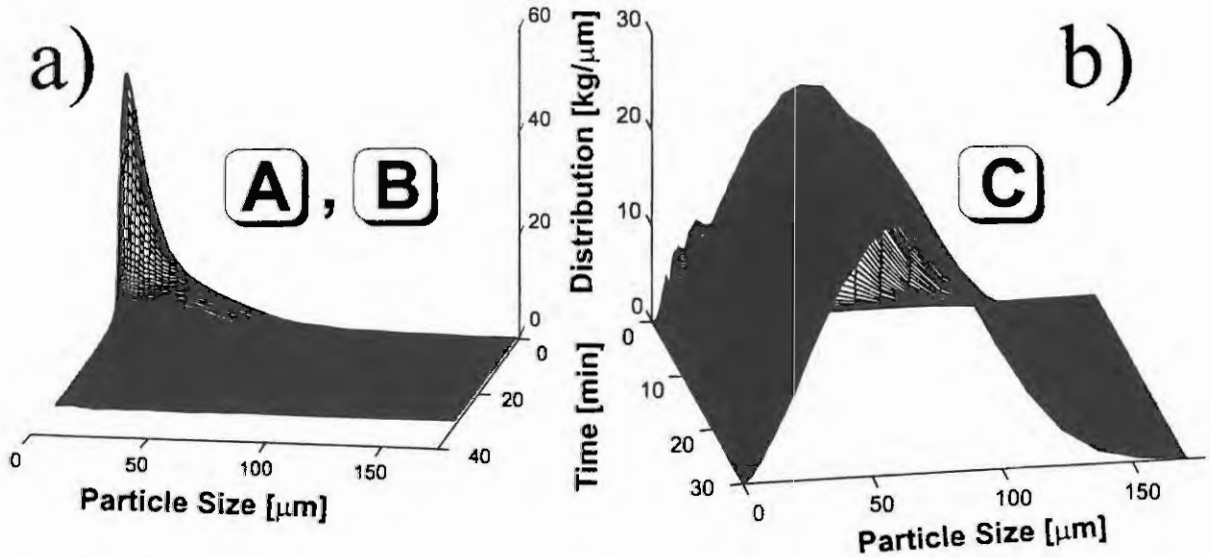


Abbildung 4: *Simulation of mechanical alloying with same parameters* $A = 0.31 \cdot 10^9/min, \alpha = 2, \beta = 4, C = 0.05$ *for all particle species A,B,C and 1-1-stoichiometry for the formation of C. a) Identical initial distribution for both powders A,B. b) Formation of the product C.*

implemented simulation tool the developed model must now be fitted to different sets of experimental data. To this end a multi-stage identification procedure is advisable which firstly characterizes the two initial powders of mechanical alloying separately from each other and then deals with the complex alloying process.

On the other hand the multidimensional distribution model sketched above is much more demanding from a computational viewpoint and an efficient numerical algorithm has to be developed to obtain reasonable computation times. To this end a reasonably class of generalized welding functions $w(a, b, c, a', b', c')$ has to be specified based on chemical and physical considerations. This is the goal of future work.

Literatur

[1] B.J.M. Aikin, T.H. Courtney, and D.R. Maurice. Reaction rates during mechanical alloying. *Materials Science and Engineering*, A147 (1991), 229–237.

[2] D. Eyre, R.C. Everson, and Q.P. Campbell. New parameterization for a discrete batch grinding equation. *Powder Technology*, 98 (1998), 265–272.

[3] M. Gao and E. Forssberg. Prediction of product size distributions for a stirred ball mill. *Powder Technology*, 84 (1995), 101–106.

[4] D. Hoppe, W. Wiechert, H. Mournier, K. Hack, D. Havart, and H. Weiß - The synthesis of nanocomposite Ag-SnO_2 contact materials by reactive high energy ball milling. In: Proc. *AMDP 99, Tokushima, Japan, Nov. 23-26, 1999*, 1999.

[5] C.C. Koch. The synthesis and structure of nanocrystalline materials produced by mechanical attrition: A review. *Nanostructured Materials*, 2 (1993), 109–129.

[6] H. Mournier. Modellierung und Simulation des Mechanischen Legierens in einer Hochenergie-Kugelmühle. Diplomarbeit, Universität Siegen, 1999.

[7] P.H. Shingu, B. Huang, S.R. Niskitani, and S. Nasu. *Suppl. to Trans. JIM*, 29 (1989), 3.

[8] K.P. Thiessen and K. Sieber. Energetische Randbedingungen trihochemischer Prozesse. Teil I-III. *Zeitschrift für Physikalische Chemie*, 260 (1979), 403–422.

[9] R. Verma and R. K. Raiamani. Environment-dependent breakage rates in ball milling. *Powder Technology*, 84 (1995), 127–137.

On a mathematical model for a problem of gas absorbtion in a liquid

Nicholas Batens and Roger Van Keer

Department of Mathematical Analysis, Faculty of Engineering, Ghent University, Belgium

Abstract. In this paper we deal with a problem of gas-liquid interactions, viz. a problem of gas absorption in a liquid by an instantaneous and irreversible reaction, taking into account a gas phase resistance. For simplicity we consider an absorption process with a single reaction. In contrast to similar problems in the literature we allow the reactor length to be large but finite and we allow a time varying unknown interface concentration of the gas component for time values $t > t^*$, t^* being the moment that the reaction plane starts moving through the liquid. This time point is determined by means of the solution of a transient diffusion problem which can be solved exactly. The evaluation of the interface concentration and the resulting mass flux requires a mathematical model for the concentration profiles in the liquid. This diffusion problem, presented in the paper is a moving or free boundary problem. We briefly indicate how this FBP can be solved numerically.

Keywords: Gas absorbtion, diffusion process, free boundary problem.

Introduction

In the chemical and processing industry absorbtion processes, in which a gaseous component dissolves in and reacts with a non volatile component in a liquid, are of primary importance. In particular, the absorption of SO_2 into aqueous alkaline solutions has received much attention because of its relevance in pollution abatement.

When gas absorption and desorption is accompanied by one or more instantaneous, irreversible chemical reactions, the liquid phase is divided in two or more zones where reacting components cannot coexist. Therefore, only diffusion of the components has to be considered in the respective regions. The reactions will take place at the boundaries of these different zones, see [5], and during the chemical absorpton process, these layers move through the liquid phase, giving rise to a free boundary problem for the evaluation of the involved concentration profiles.

In our model, we consider only one instantaneous and irreversible reaction $A + B \rightarrow C$ in the reactor, where B is the non-volatile component intially present in the reactor at a fixed concentration, and A is the gaseous component that is dissolved into the liquid. Further, as is the case for diluted gas phases, we suppose to be confronted with gas phase resistance[1]. It is assumed that, see [3] and [1], close to the liquid interface, there is a stagnant film of thickness δ through which the transport process takes place by molecular diffusion only, giving rise to a Robin-condition at the interface for the absorbed component (see further).

By the instantaneous and irreversible character of the reactions, the time interval must be divided into two subintervals. During the first interval $0 < t < t^*$ the concentration of the gaseous component at the interface remains zero, while the concentration of the liquid phase

[1] In the literature this phenomenon is referred to as gas-film resistance

reactant decreases. The time point $t = t^*$, which has to be determined, is precisely the moment at which the concentration of the B-component at the interface becomes zero and the reaction plane starts moving through the liquid. Subsequently, the concentration of both components at the reaction plane remain zero. The component A diffuses in the first zone, the component B diffuses in the second zone.

The concentration profile of the B-component has to obey a standard one dimensional BVP in the time interval $(0, t^*)$, of which the analytical solution can be obtained. The mathematical model for the two concentration profiles for $t > t^*$ is a complex 1D-free boundary problem (FBP). We also derive an auxiliary, approximate, model during a time interval $t^* < t < t^{**}$, in order to provide suitable initial data, from which a numerical approximation method may start.

The model for $0 < t < t^*$

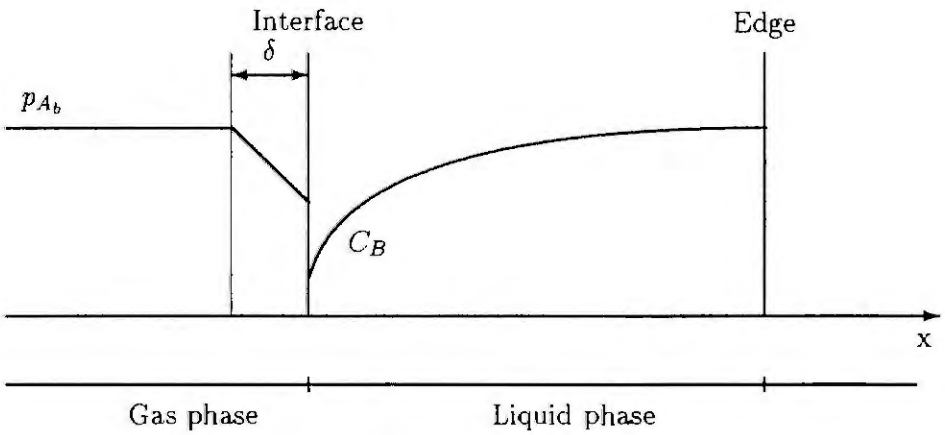


Fig. 1 Situation for $0 < t < t^*$

Up to a certain moment t^*, which is still to be determined, only diffusion of component B has to be considered while the interface concentration of component A remains zero, (cf. Figure 1). We define t^* as the moment at which the concentration of B vanishes at the interface, $C_B(0, t^*) = 0$. The mathematical model for the concentration profile $C_B(x, t)$ of the B-component in the reactor, depicted in Fig. 1, is the following BVP:

$$D_B \partial_{xx} C_B = \partial_t C_B, \qquad 0 < x < L, \quad 0 < t < t^* \tag{1}$$

$$D_B \partial_x C_B = k_g \, p_{A_b}, \qquad x = 0, \quad 0 < t < t^* \tag{2}$$

$$D_B \partial_x C_B = 0, \qquad x = L, \quad 0 < t < t^* \tag{3}$$

$$C_B(x, 0) = C_{B_b}, \qquad 0 < x < L, \quad t = 0 \tag{4}$$

Here, D_B is the diffusion coefficient of the B-component in the liquid. Moreover, kg is the gas phase mass transfer coefficient and p_{A_b} is the partial pressure of the A-component in the bulk of the gas phase. Finally, C_{B_b} represents the bulk concentration of the B-component in the liquid.

The Neumann BC at $x = 0$ corresponds to the fact that the fluxes of the components A and B are equal at the gas-liquid interface, while the BC at $x = L$ reflects the reactor edge to be impermeable for the B-component.

The analytical solution of (1-4) is found by means of Laplace transformation:

$$C_B(x,t) = C_{B_b} - \frac{k_g\, p_{A_b}\, t}{L} - k_g\, p_{A_b}\, L\Big(\frac{3\,(L-x)^2 - L^2}{6L^2} - \frac{2}{\pi^2}\sum_{n=1}^{\infty}\frac{(-1)^n}{n^2} e^{-D_B\, n^2\, \pi^2\, t\, /\, L^2}\, \cos\Big(\frac{n\,\pi\,(L-x)}{L}\Big)\Big) \quad (5)$$

Consequently, t^* will have to be obtained from the trancendental equation

$$C_{B_b} - \frac{k_g\, p_{A_b}\, t^*}{L} - k_g\, p_{A_b}\, L\Big(\frac{1}{3} - \frac{2}{\pi^2}\sum_{n=1}^{\infty}\frac{e^{-D_B\, n^2\, \pi^2\, t^*\, /\, L^2}}{n^2}\Big) = 0 \quad (6)$$

As the series $\sum_{n=1}^{\infty}\frac{x^{n^2}}{n^2}$, $0 < x \leq 1$, converges very fastly, even if x is slightly less than or equal to 1, we may approximate t^* from the equation

$$C_{B_b} - \frac{k_g\, p_{A_b}\, t^*}{L} - k_g\, p_{A_b}\, L\Big(\frac{1}{3} - \frac{2}{\pi^2}\sum_{n=1}^{N}\frac{e^{-D_B\, n^2\, \pi^2\, t^*\, /\, L^2}}{n^2}\Big) = 0 \quad (7)$$

where N is a small integer. We may set up a Newton-Rapson iteration procedure, starting with the initial guess $t^* = \max\big(\frac{3\,C_{B_b} L - k_g\, p_{A_b}\, L^2}{3\,k_g\, p_{A_b}}, 0\big)$.

A model for the initial profiles.

We consider an auxiliary, approximate, problem for the interval $t^* < t < t^{**}$, where $\Delta t = t^{**} - t^*$ is small. The position of the reaction plane at $t = t^{**}$ and the respective concentration profiles of the A- and B-components at this time point will be taken to be the initial data for the FBP for $t > t^{**}$. The situation at $t = t^*$ and the approximate situation at $t = t^{**}$ are depicted in Fig. 2-3.

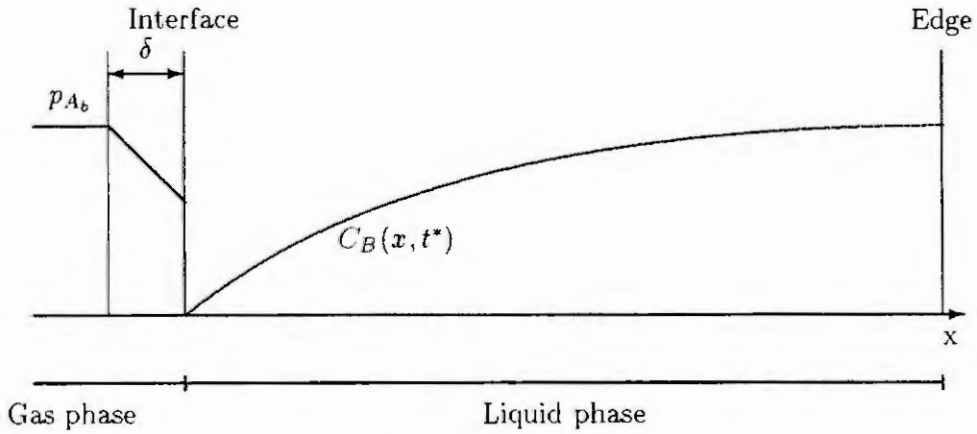


Fig. 2 Situation at $t = t^*$

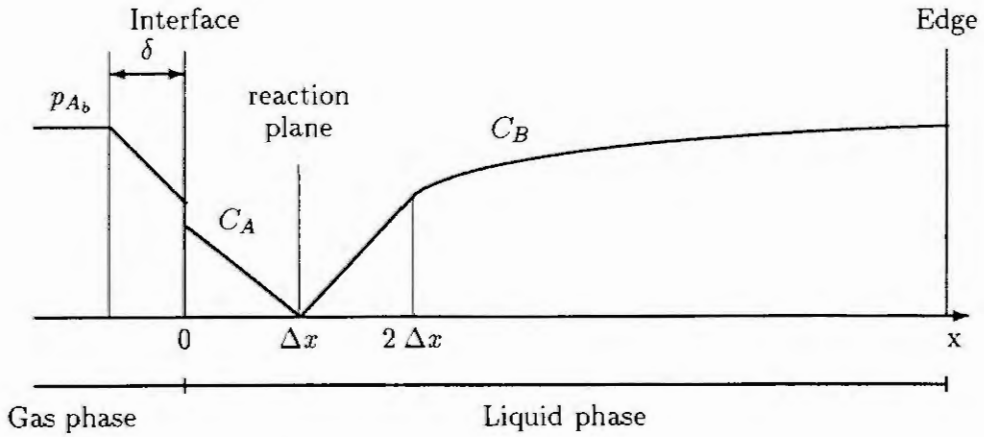


Fig. 3 Approximate situation at $t = t^{**}$

We choose a small space-interval Δx and suppose that after a small time interval Δt the reaction plane is situated in Δx, so $s(t^* + \Delta t) = \Delta x$. We further suppose that $C_B(x, t^* + \Delta t) = C_B(x, t^*)$ for $x \geq 2\,\Delta x$. Next, at $t = t^* + \Delta t$, the concentration profiles of A and B in the respective zones $0 < x < \Delta x$ and $\Delta x < x < 2\,\Delta x$ are taken to be linear functions of x, with $C_A(\Delta x, t^* + \Delta t) = C_B(\Delta x, t^* + \Delta t) = 0$, $C_B(2\,\Delta x, t^* + \Delta t) = C_B(2\,\Delta x, t^*)$, see Fig. 2 and Fig. 3. The interface concentration $C_A(0, t^* + \Delta t)$ has to be determined.

The total amount of material B that has reacted away in the time interval $(t^*, t^* + \Delta t)$ is

$$R = \int_0^{2\,\Delta x} C_B(c, t^*)dx - \frac{\Delta x\, C_B(2\,\Delta x, t^*)}{2} \tag{8}$$

Consequently, the same amount of component A has reacted away in this time interval. Hence, the total amount of A that has entered the reactor during this time interval is equal to

$$R + \frac{\Delta x\, C_A(0, t^* + \Delta t)}{2} \tag{9}$$

Imposing the equality of the flux of the A and B-components at the reaction plane we get

$$C_A(0, t^* + \Delta t) = \frac{D_B}{D_A} C_B(2\,\Delta x, t^*) \tag{10}$$

Finally, to relate Δt to Δx, we assume a constant flux F_A of component A at the interface during the small time interval $(t^*, t^* + \Delta t)$, viz the average of the flux at $t = t^*$ and the one at $t = t^* + \Delta t$. From (2) the former flux is

$$D_A C_A(0, t^*) = -k_g\, p_{A_b}, \tag{11}$$

The flux at $t = t^* + \Delta t$ reads

$$D_A C_A(0, t^* + \Delta t) = -k_b\big(p_{A_b} - H\, C_A(0, t^* + \Delta t)\big), \tag{12}$$

where H is Henry' constant, relating the unknown pressure of the A-component at the gas side of the interface with its equilibrium concentration at the liquid side. Thus we take,

$$F_A = -k_b\big(p_{A_b} - H\, \frac{C_A(0, t^* + \Delta t)}{2}\big) \tag{13}$$

Noticing that $F_A\, \Delta t$ is exactly the total amount of component A that has entered the reactor during the interval $(t^*, t^* + \Delta t)$, which is given by (9), we arrive at Δt in terms of Δx, when still taking into account (10).

The moving diffusion problem for $t > t^{**}$

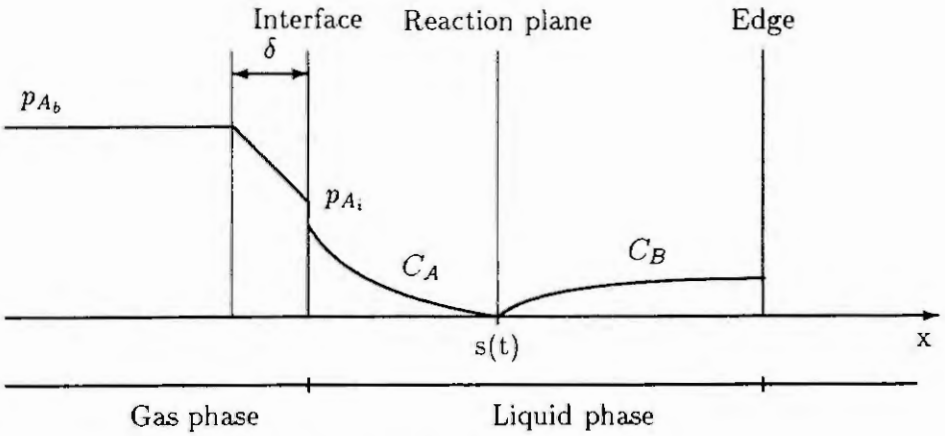


Fig. 4 Situation for $t > t^{**}$

From $t = t^*$ onwards, the reaction plane $x = s(t)$ started moving through the liquid. The typical situation is shown in Fig. 4.
The two component diffusion problem for $t > t^{**}$ can be described by a FBP in 1D for the concentrations $C_A(x,t)$ and $C_B(x,t)$, consisting of the two governing DEs,

$$\begin{aligned} D_A \partial_{xx} C_A &= \partial_t C_A, && 0 < x < s(t), \quad t > t^{**}, \\ D_B \partial_{xx} C_B &= \partial_t C_B, && s(t) < x < L, \quad t > t^{**}, \end{aligned} \tag{14}$$

along with the respective BCs,

$$D_A \partial_x C_A = -k_g (p_{A_b} - H\, C_A), \qquad x = 0, \quad t > t^{**}, \tag{15}$$

$$D_B \partial_x C_B = 0, \qquad x = L, \quad t > t^{**}, \tag{16}$$

and along with transmission conditions (TCs)

$$C_A = C_B = 0, \qquad x = s(t), \quad t > t^{**} \tag{17}$$

$$-D_A \partial_x C_A = D_B \partial_x C_B, \qquad x = s(t), \quad t > t^{**} \tag{18}$$

and starting from the initial data,

$$\begin{aligned} s(t^{**}) &= \Delta x \\ C_A(x, t^{**}) &= f_A(x) && 0 \le x \le \Delta x, \\ C_B(x, t^{**}) &= f_B(x) && \Delta x \le x \le L \end{aligned} \tag{19}$$

Here, D_B, k_g, p_{A_b} and H have a similar meaning as above, while D_A stands for the diffusion coefficient of component A in the liquid phase. Furthermore, the *initial* profiles $f_A(x)$ and $f_B(x)$ are the concentration profiles at $t = t^* + \Delta t$, obtained in the previous section.

The TC (17) reflects the fact that components A and B cannot coexist, while (18) ensures the diffusion fluxes of the reactants to satisfy the stoichiometry of the reaction. Finally, the BC (15) arises from the gas phase resistance at the gas-liquid interface as mentioned above, while (16) expresses the impermeability of the reactor edge.

On the numerical solution

We now briefly comment on a numerical method for the FBP (14-19), starting from the evaluated initial data at $t = t^{**}$. A detailed outline of the method, even for the case of two moving reaction planes, and a numerical example can be found in [2].

First, a fixed domain transformation is performed, mapping each zone onto the interval $[0, 1]$ to eliminate the moving character of the internal boundary. This results in a highly *nonlinear*, *nonlocal* BVP, the underlying DEs of which now explicitely contain $s(t)$ and $\dot{s}(t)$.

Secondly, having deliberately choosen a fixed (nonuniform) grid for each of the zones, the new DEs are discretised by means of a nonstandard finite difference method, to obtain a system of difference-differential equations (DDEs) for the concentration profiles at the grid points as functions of time.

Using the transformed version of (18), it is possible to suitably approximate $s(t)$ as a function of the concentrations of the reacting components in gridpoints near the moving reaction plane and, similarly, to approximate $\dot{s}(t)$ as a function of the same concentrations, but also *linearly* depending on the time-derivatives of these concentrations. This allows us to eliminate $s(t)$ and $\dot{s}(t)$ from the DDEs.

This leads us to a highly nonlinear initial value problem (IVP) for a system of 1 st order ODEs, for the time continuous approximate concentration profiles at the relevant grid points. Here the initial data at $t = t^{**}$ are evaluated by the fixed domain transformation of the initial profiles $f_A(x)$ and $f_B(x)$, determined above. By manipulation of the system of ODEs, taking profit of the structure of the coefficient matrix, the IVP can be reduced to a standard form

$$\begin{aligned} &\dot{X}(t) = F(t, X(t)), \qquad t > t^{**}, \\ &X(t^{**}) = G, \end{aligned}$$

where $X(t)$ is the vector of the values of the concentration profiles of the A- and B-component, arranged in a suitable order.

This IVP can be solved numerically by an Adams-Moulton Adams-Bashforth predictor corrector method, [6]. As an example of an advanced computer package we may refer to [4].

References

[1] D. Baetens, R. Van Keer, J. Hosten, Gas-liquid reaction: absorption accompanied by an instantaneous chemical reaction, in: R. Van Keer, C.A. Brebbia, Moving Boundaries *IV*, Computational Mechanics Publications, Southampton, 1997, pp. 185-195

[2] N. Batens, R. Van Keer, *A numerical method for a free boundary problem arising from chemical kinetics*, Journ. Comp. Appl. Math. **111** (1999), 187-199

[3] G. Froment, K. Bischoff, Chemical reactor analysis and design, Wiley, New York, 1990

[4] A. Hindmarsh, Odepack, a systematized collection of odesolvers, in: R.S. Stepleman et al. (eds.), Scientific computing, North-Holland, Amsterdam, 1983, pp. 55-64.

[5] K. Onda, T. Kobayashi, M. Fujine, M. Takahashi, Behavior of the reactionplane movement in gas absorbtion accompanied by instantaneous chemical reactions, SIAM Chem. Eng. Sc. **26** (1971) 2009-2026.

[6] Stoer, J. Bulirsch, R., *Introduction to numerical analysis*, Springer, New York, 1993

MODELLING OF SYSTEMS WITH EQUILIBRIUM REACTIONS

M.R.Westerweele, M. Akhssay and H.A. Preisig
Systems and Control Group, Eindhoven University of Technology
P.O.Box 513, 5600 MB Eindhoven, The Netherlands
M.R.Westerweele@tue.nl

Abstract. This paper presents a part of our research on the mathematical modelling of processes. It concerns the problems that can arise when the elementary systems (lumps) that constitute the model of the process contain equilibrium reactions. When equilibrium reactions are considered, the kinetic equations, which describe the dynamics of the reactions, are often omitted, because the equilibrium reactions are assumed to have very fast dynamics relative to the time scale of the process. This results in a high-index DAE for the model, because the reaction rates of these reactions occur only in the differential equations. By making linear combinations of the component massbalances one can eliminate the undefined production terms and consequently reduce the dimension of the original system.

Declaration of Symbols

$\underline{\mathbf{n}}$	:= species (component) mass vector in moles
$\underline{\underline{\hat{\mathbf{N}}}}$	:= molar mass flow matrix := $(\underline{\hat{\mathbf{n}}}_1\ \underline{\hat{\mathbf{n}}}_2\ ....\underline{\hat{\mathbf{n}}}_m\ )$, with $\underline{\hat{\mathbf{n}}}_m$ = molar mass flow m
$\underline{\boldsymbol{\alpha}}$	:= vector of unit direction of reference co-ordinates, with $\alpha_i \in \{-1, +1\}$
$\underline{\tilde{\mathbf{n}}}$	:= procuction rate for all species
$\underline{\mathbf{A}}$	:= species vector, containing all 'reactive' species of the system under consideration $= (A_1\ A_2\ .....A_n)^T$ for n species
$\tilde{\xi}_{r,A_j}$	:= normalized rate of formation of species A_j by reaction r
$\underline{\tilde{\boldsymbol{\xi}}}$	:= normalized reaction rates vector = $(\tilde{\xi}_1\ \tilde{\xi}_2\ .....\tilde{\xi}_m)^T$ for m reactions
$\underline{\underline{\mathbf{S}}}$	:= stoichiometric matrix = $(\underline{\nu}_1\ \underline{\nu}_2.....\underline{\nu}_m)$ for m reactions. (dimension: n x m)
$\underline{\underline{\mathbf{S}}}_{eq}$	:= part of the stoichiometric coefficient matrix $\underline{\underline{\mathbf{S}}}$ that pertains to the equilibrium reactions
$\underline{\underline{\mathbf{S}}}_{r}$	:= part of the stoichiometric coefficient matrix $\underline{\underline{\mathbf{S}}}$ that pertains to the remaining reactions

Introduction

We are concerned with the mathematical modelling of macroscopic (bio-) chemical processes as they appear in general when modelling chemical or biological plants. The ultimate goal of our research is to implement a structured modelling methodology in a set of computer programs, the first of which is the Modelle, all aiming at effectively assisting in the development of process models.

A (mathematical) model of a process is usually a system of mathematical equations, whose solutions reflect certain quantitative aspects (dynamic or static behaviour) of the process to be modelled. More often than not, the time spent on collecting the information necessary to properly define such a model is much greater than the time spent by a simulator program in finding a solution. During the last decades the demand for models is increasing fast. At the same time, the complexity of the models also increases, which makes the model construction even more time consuming and error-prone. Moreover, there are many different ways to model a process: different time scales, different levels of detail, different assumptions, different interpretations of (different parts of) the process, etc. Thus a vast number of different models can be generated for the same process.

All this calls for a systematization of the modelling process, comprising of an appropriate, well-structured modelling methodology for the efficient development of adequate, sound and consistent process models. Modelling tools building on such a systematic approach support teamwork, re-use of models, provide complete and consistent documentation and, not at least, improve process understanding and provide a foundation for the education of process technology.

It is generally known that often high-index DAEs arise in process models due to "simplifications" that impose additional constraints on the differential variables. For example, the assumptions of phase equilibrium, reaction equilibrium, negligible dynamics or steady state of parts of the process can lead to high-index DAEs. In all cases, a high-index DAE implies that the differential variables in the model are not independent and cannot all be assigned arbitrary initial values. The general approach to solve

the problem of the high index is to either change the assumptions such that no high-index problem arises or reduce the index by differentiating some of the equations a sufficient number of times. This is, however, not always necessary because sometimes it is possible to prevent the high index from occurring by applying algebraic approaches that utilise modelling insight and respective algebraic procedures.

This paper presents a part of our research on the mathematical modelling of processes. It concerns the problems that can arise when the elementary systems (lumps) that constitute the model of the process contain equilibrium reactions. The problem we present here is not a new one (e.g.:[1], [2], [3], [4]) nor is the solution in its basic thought, though new is the generality with which the problem is solved and thus the extensive width of problems that fall into this category. We present the problem in a very general form such that it can be applied for virtually every (bio-) chemical process, without the need for any additional (constraining) assumptions (such as: constant volume, constant density, constant temperature, etc.). The approach taken can be seen as an application of the simple index reduction algorithm as presented in [3]

Modelling of Reactive Systems

The behaviour of a system is characterised by the evolution of its state with time. The choice of the elementary system is driven by the choice of the time-scale in which the dynamics of the system is to be resolved. This choice is fundamental to the analysis.

The natural state of a system can be described a set of fundamental extensive quantities. These fundamental extensive variables represent the "extent" of the system c.q. the quantities being conserved in the system. The conservation principles describe the fundamental dynamics of an elementary system. In modelling reactive systems, the conservation of component masses is the obvious choice of fundamental extensive quantities. Their accumulation in the system is balanced by the transfer across the system boundary and the internal conversion through reaction:

$$\dot{\underline{n}} = \underline{\underline{\hat{N}}}\,\underline{\alpha} + \tilde{\underline{n}} \tag{1}$$

The production term $\tilde{\underline{n}}$ may be written in the form:

$$\tilde{\underline{n}} = \underline{\underline{S}}^T \tilde{\underline{\xi}} \tag{2}$$

where $\underline{\underline{S}}$, the stoichiometric matrix, describes the molar balances of the reactions only. Each row in this matrix gives information of the molar equivalent reacting (negative integer equivalent) and being produced (positive integer equivalent). With $\underline{A}$ being the species vector, the reactions are:

$$\underline{\underline{S}}\,\underline{A} = \underline{0} \tag{3}$$

The production term sums for each species the effects of each of the reactions. For each reaction, a quantity $\tilde{\xi}$ can be defined that is a normalized rate of reaction and which can be shown to be the time derivative of the extent of reaction.

Modelling systems in a range of time scales, the capacity terms are chosen accordingly but also the transport and the production terms. For parts being outside of the time-scale in which the dynamics are being modelled, a pseudo-steady state assumption is being made. For example, (very) fast reactions - fast in the measure of the considered range of time scale - are assumed to reach the equilibrium (for all practical purposes) instantaneously, and very slow ones do not appreciatly occur and may be simply ignored. For the effects of small and large capacities, the singular perturbation theory is applicable. For the reaction, we split the set of reactions into three parts, namely the very slow ones for which no reaction occurs, one that occur in the time-scale of interest and one part which is very fast and for which we assume the equilibrium to be reached instantaneously, this all in an elementary system defined as a single, uniform phase piece of the spatial domain. As the non-reactive parts do not further contribute to the discussion, they are left out iu the sequel. With the two parts being labelled as *eq* for the fast part and r for the parts with dynamics in the relevant time scale, the production term (2) reads:

$$\tilde{\underline{n}} = \underline{\underline{S}}^T \tilde{\underline{\xi}} = \underline{\underline{S}}_{eq}^T \tilde{\underline{\xi}}_{eq} + \underline{\underline{S}}_r^T \tilde{\underline{\xi}}_r \tag{4}$$

Consequently, the component mass balances will take the following form:

$$\dot{\underline{n}} = \hat{\underline{\underline{N}}}\,\underline{\alpha} + \tilde{\underline{n}} = \hat{\underline{\underline{N}}}\,\underline{\alpha} + \underline{\underline{S}}_{eq}^{T}\tilde{\underline{\xi}}_{eq} + \underline{\underline{S}}_{r}^{T}\tilde{\underline{\xi}}_{r} \tag{5}$$

The normalized reaction rates $\tilde{\underline{\xi}}_r$ of the reactions in the relevant time scale must be defined by kinetic rate equations. The remaining fast (equilibrium) reaction rates $\tilde{\underline{\xi}}_{eq}$ are not defined, because the equilibrium reactions are considered to have very fast dynamics relative to the time scale of the process. For these reactions only the reaction outcome has to be given in the form of an equilibrium relation, which should hold at every time instant. This is usually a nonlinear, algebraic relation that relates the masses of the involving species to each other (e.g.: $K = \frac{c_A}{c_B c_c}$). Consequently, unlike the situation where no equilibrium reactions occur, the initial values of the masses of the involved species cannot be arbitrarily chosen because the quantities of some species in the system are now directly related to the quantities of some other species in the system. This results in some differential equations of the system being directly related to each other, which reduces the actual dimension of the system. Also, the production terms $\tilde{\underline{\xi}}_{eq}$ of the equilibrium reactions occur only in the component mass balances and cannot be determined directly from system equations. Since these terms are not really of interest anyhow, we can resolve the problems by eliminating the undefined production terms $\tilde{\underline{\xi}}_{eq}$ by forming linear combinations of the component mass balances. This can be achieved by multiplying (5) with a matrix $\underline{\underline{\Omega}}$, of which the rows constitute a basis for the nullspace of $\underline{\underline{S}}_{eq}^{T}$, (i.e:.$\underline{\underline{\Omega}}\,\underline{\underline{S}}_{eq} = \underline{\underline{0}}$). This results in:

$$\begin{aligned} \underline{\underline{\Omega}}\,\dot{\underline{n}} &= \underline{\underline{\Omega}}\,\hat{\underline{\underline{N}}}\,\underline{\alpha} + \underline{\underline{\Omega}}\,\underline{\underline{S}}_{eq}^{T}\tilde{\underline{\xi}}_{eq} + \underline{\underline{\Omega}}\,\underline{\underline{S}}_{r}^{T}\tilde{\underline{\xi}}_{r} \\ &= \underline{\underline{\Omega}}\,\hat{\underline{\underline{N}}}\,\underline{\alpha} + \underline{\underline{\Omega}}\,\underline{\underline{S}}_{r}^{T}\tilde{\underline{\xi}}_{r} \end{aligned} \tag{6}$$

If we now define new variables:

$$\underline{n}^{*} = \underline{\underline{\Omega}}\,\underline{n} \tag{7}$$

the condensed representation, which is minimal, results:

$$\dot{\underline{n}}^{*} = \underline{\underline{\Omega}}\,\hat{\underline{\underline{N}}}\,\underline{\alpha} + \underline{\underline{\Omega}}\,\underline{\underline{S}}_{r}^{T}\tilde{\underline{\xi}}_{r} \tag{8}$$

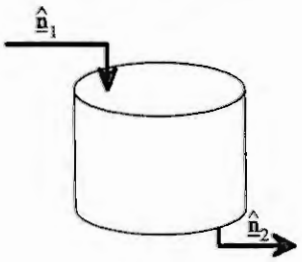
$$\underline{K} = \underline{f}(\underline{n}) \tag{9}$$

By multiplying the component massbalances (5) by $\underline{\underline{\Omega}}$ and introducing the algebraic relations (9) which describe the outcome of the equilibrium reactions, we have reduced the dimension of the original system and eliminated the undefined production terms. So, for a system which consists of n components among which k independent equilibrium reactions are occuring, the condensed representation of the system comprises $(n-k)$ differential equations (8) and k algebraic equations (9), which describe the dynamics of the system.

This is a semi-explicit DAE system of index one which can be solved with standard algorithms. The only differential variables are the terms $\underline{n}^{*}$, so if we specify $\underline{n}^{*}(0)$ and properly define all additional variables (such as $\hat{\underline{\underline{N}}}$, $\underline{\alpha}$, $\underline{\underline{S}}_{eq}$, $\underline{\underline{S}}_{r}$ and $\tilde{\underline{\xi}}_{r}$), all quantities $\underline{n}$ of all species can be calculated.

Example

As an example we consider a CSTR with 6 components A, B, C, D, E and F. The reactor has a constant volumetric in- and outflow $\hat{V}$ and there are three reactions taking place in this reactor. Two of these reactions (the first and the third) are considered to equilibrium reactions.



$$\begin{aligned} A &\rightleftarrows B + D \quad K_1 = \tfrac{c_B c_D}{c_A} \\ B &\rightleftarrows C + D \quad \tilde{\xi}_2 = (k_{2a} c_B - k_{2b} c_C c_D) V \\ D + E &\rightleftarrows F \quad K_3 = c_D c_E \end{aligned}$$

Normally one would expect that the dimension of the state space of the system equals the number of components which are associated with the system (when energy of the system is not modelled!) and that the model of the system would include 6 differential equations (one for each component). But since two of the occurring reactions are considered to be equilibrium reactions, two of those differential equations are replaced by algebraic relationships and the actual dimension of the state space of the system is reduced to four:

$$\underline{\underline{\mathbf{S}}}_{eq}^T = \begin{bmatrix} -1 & 0 \\ 1 & 0 \\ 0 & 0 \\ 1 & -1 \\ 0 & -1 \\ 0 & 1 \end{bmatrix}; \quad \underline{\underline{\mathbf{S}}}_{r}^T = \begin{bmatrix} 0 \\ -1 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}; \quad \tilde{\xi}_r = \tilde{\xi}_2; \quad \underline{\underline{\Omega}} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & -1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix};$$

$$\underline{\alpha} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}; \quad \hat{\underline{\underline{\mathbf{N}}}} = \begin{bmatrix} \hat{\underline{\mathbf{n}}}_1 & \hat{\underline{\mathbf{n}}}_2 \end{bmatrix} = \begin{bmatrix} \underline{\mathbf{c}}_1 \hat{V} & \underline{\mathbf{c}} \hat{V} \end{bmatrix}; \quad \underline{\mathbf{c}} = \frac{\underline{\mathbf{n}}}{V};$$

The model of the system can be written as follows:

$$\begin{aligned} \underline{\underline{\Omega}}\,\dot{\underline{\mathbf{n}}} &= \underline{\underline{\Omega}}\,\hat{\underline{\underline{\mathbf{N}}}}\,\underline{\alpha} + \underline{\underline{\Omega}}\,\underline{\underline{\mathbf{S}}}_r^T \tilde{\underline{\xi}}_r \\ K_1 &= \frac{c_B c_D}{c_A} \\ K_3 &= c_D c_E \end{aligned}$$

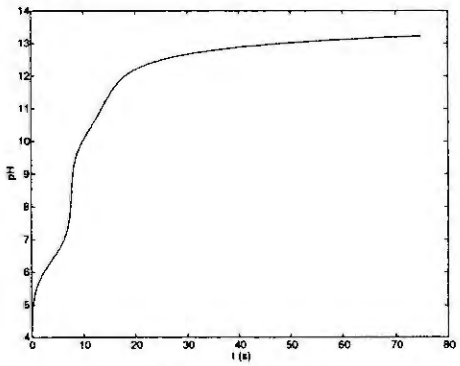


Figure 1: pH of the contents of the CSTR as funcion of the elapsed time.

This model can be easily solved with a DAE solver when the matrix $\hat{\underline{\underline{\mathbf{N}}}}$ is appropriately defined and appropriate initial conditions are given. Figure 1 is the result of a simulation of the system (with $A = H_2CO_3, B = HCO_3^-, C = CO_3^{2-}, D = H^+, E = OH^-, F = H_2O$)

Conclusions

This method of calculation has been possible only because we did not insist on determining *all* the variables in the original system; in particular, we decided that r was of no interest and we eliminated it using purely algebraic (i.e. no differentiations) manipulations.

In doing so, we have reduced the dimension of the original system and introduced the new variables $\underline{\mathbf{n}}^*$ defined by equation (7). $\underline{\mathbf{n}}^*$ Have an interesting physical interpretation: They are quantities that remain unchanged by the equilibrium reactions and are called "reaction invariants". Equation (8) can be interpreted as the balances of these quantities and, as might be expected, these balances do not involve reaction terms of equilibrium reactions.

References

[1] Fjeld M. Asbjørnsen O.A. Response modes of continuous stirred tank reactors. *Chem. Engng. Sci.*, 25:1627–1636, 1970.

[2] Håvard I. Moe, S. Hauan, K.M. Lien, and T. Hertzberg. Dynamic model of a system with phase- and reaction equilibrium. *Comput.Chem.Eng.*, 19:S513–S518, 1995.

[3] Håvard I. Moe. *Dynamic process simulation: studies on modeling and index reduction.* PhD thesis, University of Trondheim, 1995.

[4] C.C. Pantelides. Dynamic behaviour of process systems. Technical report, Centre for Process Systems Engineering, Imperial College, London, 1998.

MODELING AND COMPUTATIONALLY EFFICIENT SIMULATION OF CHROMATOGRAPHIC SEPARATION PROCESSES

Karsten-Ulrich Klatt, Guido Dünnebier and Sebastian Engell
Process Control Laboratory, Department of Chemical Engineering
University of Dortmund, D-44221 Dortmund, Germany
Email: k.klatt@ct.uni-dortmund.de

Abstract. Chromatographic separation processes are an emerging technology, especially in the field of fine chemicals and pharmaceutical products. As an alternative to conventional batch chromatography, the simulated moving bed (SMB) process gained more and more impact recently. Because of the complex dynamics, the automatic control of chromatographic separation processes is a challenging task, the solution of which requires reliable and computationally efficient simulation models. The purpose of this contribution is to give an overview about the generation of dynamic models both for single chromatographic columns and SMB processes. We here distinguish chromatographic processes by the type of adsorption isotherm and proceed from the ideal model for chromatographic columns while increasing the model complexity as far as necessary in order to build a simulation model which on the one hand is computationally effective and on the other hand correctly describes the dynamics of the process which is relevant for on-line optimization and control purposes.

Introduction

Chromatographic separation processes provide a powerful tool for the separation of mixtures in which the components have different adsorption affinities, especially when high resolutions and purities are required and/or when conventional thermal unit operations like distillation are not desirable due to the thermal instability of the substances involved. Typical applications are found in the pharmaceutical industry and in the production of other life science products. The different components are separated by their different speed of propagation in a two phase system. There exist two different modes of operation in preparative scale separations. Batch chromatography is today's standard operating mode but is a discontinuous process resulting in highly diluted products. To increase the separating power of the system, a continuous countercurrent operation is desirable, but the real countercurrent of the solid leads to serious operating problems. In this context, the Simulated-Moving-Bed (SMB) technology is becoming an important technique for large scale continuous chromatographic separation processes. The SMB process is realized by connecting several single chromatographic columns in series. The countercurrent movement is approximated by a cyclic switching of the inlet and outlet ports in the direction of the fluid stream. Thus, this process shows complex mixed continuous and discrete dynamics.

Because of the complex dynamics, optimal operation and control of chromatographic separation processes with regard to safe and economical operation and meeting the product specifications at any time is a very challenging task. For this, accurate and reliable dynamic models are necessary which include the continuous dynamics of the single columns as well as the discrete events resulting from the cyclic operation. The purpose of this contribution is to expose a framework for the modeling and simulation of chromatographic separation processes, incorporating recent own results and highlighting new numerical approaches. Because the ultimate goal is to use the generated simulation models for model-based control, computational efficiency is a crucial issue. We propose a modular approach where the simulation model of the more complex SMB process is assembled by connecting dynamic models for the single columns incorporating the cyclic port switching. This modular approach allows the use of any dynamic column model required and utilizes the same mathematical access to the generation of the simulation model for batch processes and SMB processes as well. Because of the fact that there is increasing interest in the more complex SMB process, we focus in this condensed exposition on generating simulation

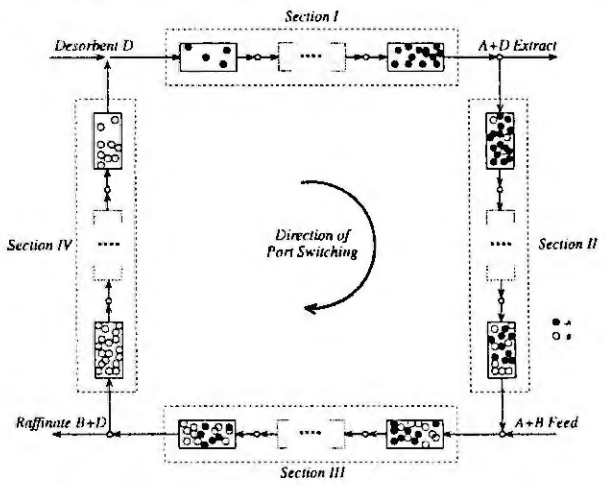


Figure 1: Scheme of a SMB process

models for simulated moving bed chromatography processes. The next section supplies a brief introduction into this certain type of process.

The Simulated Moving Bed Process

The simulated moving bed chromatographic process is the technical realization of a countercurrent adsorption process, approximating the countercurrent flow by a cyclic port switching. It consists of a certain number of chromatographic columns in series (see fig. 1) while the countercurrent movement is achieved by sequentially switching the inlet and outlet ports one column downwards in the direction of the liquid flow after a certain time period τ. In the limit of an infinite number of columns and infinitely short switching periods, the SMB operating mode comprises a real countercurrent process. By means of the relative position of the columns to the feed and draw-off nodes, the process can be divided into four different sections. The flow rates are different in every section and each section has a certain function in the separation of the mixture. The actual separation is performed in the two central sections where component B is desorbed and component A is adsorbed. Component A is desorbed in the first section to regenerate the adsorbent, and component B is adsorbed in the fourth section to regenerate the desorbent. The net flow rates of the components have different signs in the central sections II and III, thus components B and A are transported from the feed inlet upstream to the raffinate outlet with the fluid stream and downstream to the extract outlet with the "solid stream", respectively.

For economic reasons, a small number of columns is often desirable. The stationary regime of this process is a cyclic steady-state (CSS), in which in each section an identical transient during each period between two valve switches takes place. The CSS state is practically reached after a certain number of valve switches, but the system states are still varying over time because of the periodic movement of the inlet and outlet ports along the columns.

Modeling and Simulation

The mathematical modeling of single chromatographic columns has been extensively described in the literature by several authors (see [1] for a recent review). From a mathematical point of view, it is useful to distinguish chromatographic processes by the type of adsorption isotherms. The type of isotherm describing the thermodynamic behavior of the modeled system influences the structure of the resulting mathematical problem substantially. Processes with linear or simple Langmuir isotherms lead to systems of uncoupled differential equations which are easier to solve than those with coupled non-linear adsorption behavior, e.g. competitive Langmuir and Bi-Langmuir isotherms. Moreover, the modeling approaches can be classified by the physical phenomena they include and thus by their level of complexity. Fig. 2 shows this classification schematically.

There are two different approaches to built a mathematical model for the SMB process. The one tries to approximate the SMB process by the equivalent continuous true moving bed process (TMB model). The TMB model is fairly suitable for describing the steady-state operation of processes with three ore more columns in each section [2], but for processes with fewer columns (which are more and more utilized in industrial applications in order to reduce the investment costs) the conformity becomes poor. Furthermore, the TMB model does not describe the dynamics of the process which is mandatory for model-based control purposes. Therefore, we here follow the second approach which is building a realistic model of the SMB process by connecting dynamic models of the single chromatographic columns while considering the cyclic port switching. In this context, the respective fluid velocities and the inlet concentrations for each section are calculated by mass balances around the inlet and outlet nodes (node model, see [3]).

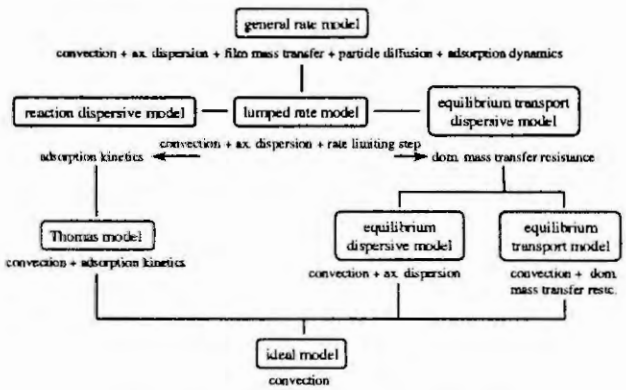


Figure 2: Classification of column models

Most dynamic SMB models reported in the literature so far use an equilibrium transport dispersive column model. It is based on the adsorption equilibrium isotherm and a linear driving force approach for the mass transfer from bulk to solid phase. This results in a set of PDEs for the bulk phase and ODEs for the solid phase. They use finite difference, finite element or collocation methods to solve the system of model equations (see e.g. [2];

[4] and [5]). The computation times of these approaches are often within the range of the real process time, which is several minutes per switching period. Thus, they are not well suited for on-line optimization and control applications. The purpose of the work presented here was to generate computationally more efficient simulation models by a bottom up strategy for the single column models. We proceed from the ideal model which only includes convection and adsorption phenomena and increase the complexity as far as necessary to achieve sufficient accuracy.

The Ideal Model for Linear Isotherms

Neglecting both mass transfer resistance and axial dispersion results in the following balance equation for the bulk phase concentration of each component

$$\frac{\partial c_i}{\partial t} + \frac{1-\varepsilon}{\varepsilon}\frac{\partial q_i}{\partial t} = -u\frac{\partial c_i}{\partial x} \quad , \tag{1}$$

where u is the fluid velocity in each section of the process. The solution of this hyperbolic PDE yields in the case of a linear adsorption isotherm ($q = K \cdot c$) a constant speed of propagation for each single species

$$w_i = \frac{u}{1 + K_i (1-\varepsilon)/\varepsilon} \quad . \tag{2}$$

Based on this expression, a dynamic SMB model was proposed in [6] which only contains algebraic equations to calculate the location of the adsorption and desorption fronts over time. The computational time is almost negligible. However, from fig. 3 it is obvious that this model gives an insight into the qualitative behavior but shows unrealistic shock fronts and is not accurate enough w.r.t. the crucial control variables (product concentrations).

The Dispersive Model for Linear Isotherms

In order to overcome the shortcomings of the ideal model we include mass transfer resistance and axial dispersion in the next step. In [7] it was shown that in case of a linear isotherm these two effects are additive and can be incorporated into a single parameter, the apparent dispersion coefficient D_{ap}. This results in a quasi-linear parabolic partial differential equation for each species:

$$(1 + k_i')\frac{\partial c_i}{\partial t} + u\frac{\partial c_i}{\partial x} = D_{ap}\frac{\partial^2 c_i}{\partial x^2} \quad . \tag{3}$$

A closed form solution of this type of equation for a set of general initial and boundary equations by double Laplace transform can be found in [8]. From this, we generate the dynamic SMB model by connecting the solution for each single column by the respective node model (see [9] for details of the implementation). Fig. 3 shows the simulation results for a sugar separation in an 8 column SMB plant. The DLI model shows promising correspondence with the more complex LDF simulation model proposed in [4] which has been experimentally verified, while the computation times could be reduced by two orders of magnitude.

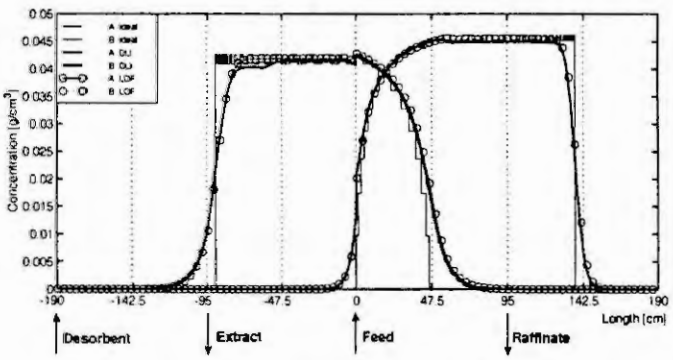


Figure 3: Axial concentration profile at CSS (Fructose/Glucose)

The Nonlinear Wave Model

Neglecting all non-ideal effects but assuming nonlinear adsorption equilibrium, the balance equations are similar to eqn. (1), but because of the competitive adsorption isotherms the hyperbolic PDEs are now strongly coupled. In the case of competitive Langmuir isotherms this system can be solved by using the theory of nonlinear wave propagation. The solution for standard Riemann problems as they occur in batch operation has been widely documented in the literature. Because of the more general initial and boundary conditions in the SMB operating mode, we extended this solution procedure by incorporating the front tracking approach [10] (see [11] for details of the implementation). This results in a computationally very efficient simulation model which however shows the same lack of accuracy as the ideal model for the linear case (see fig. 4).

The General Rate Model

In order to generate both an accurate and computationally efficient dynamic model also in the case of general nonlinear adsorption isotherms we first followed the same approach as in the linear case by lumping the non-idealities into a single parameter. However, a closed-form solution is no longer possible in the nonlinear case and the numerical solution using standard techniques for the spatial discretization did not improve the computational efficiency substantially. Fortunately, there exists a very effective numerical solution for the complex general rate model

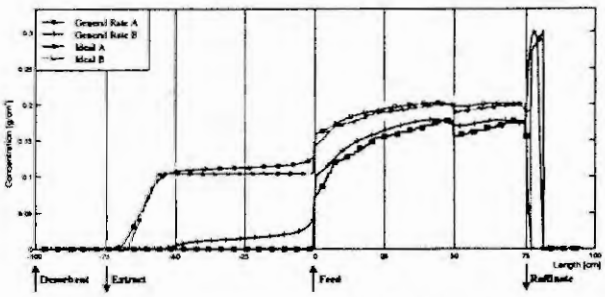


Figure 4: Axial concentration profile at CSS (Aromatics)

$$\frac{\partial c_i}{\partial t} = D_{ax} \cdot \frac{\partial^2 c_i}{\partial x^2} - u\frac{\partial c_i}{\partial x} - \frac{3(1-\varepsilon)k_{f,i}}{\varepsilon r_p}\left(c_i - c_{pi}(r_p)\right) \quad ; \quad (1-\varepsilon_p)\frac{\partial q_i}{\partial t} + \varepsilon_p \frac{\partial c_{pi}}{\partial t} = \varepsilon_p D_{p,i}\left[\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2 \frac{\partial c_{pi}}{\partial r}\right)\right], \quad (4)$$

incorporating arbitrary nonlinear isotherms, proposed in [12] for a single batch column. A finite element formulation is used to discretize the convection-dominated parabolic PDEs of the fluid phase, and orthogonal collocation for the solid phase PDEs. We extended this approach to SMB processes and the simulation results are completely consistent with those of the complex dynamic SMB models reported in the literature so far while again reducing the computational time by almost two orders of magnitude. Fig. 4 shows the simulation results for a SMB aromatics separation compared to the wave theory model.

Conclusions

We have presented a modular approach for the simulation of chromatographic separation processes examining dynamic column models of different levels of complexity. The dispersive model for linear isotherms and the particular implementation of the general rate model represent very efficient and accurate dynamic simulation models. Their implementation in an on-line optimization and control scheme is subject of our current work.

References

[1] Guiochon, G., G. Golshan-Shirazi and A.M. Katti: *Fundamentals of Preparative and Nonlinear Chromatography*. Academic Press, Boston, 1994.

[2] Lu, Z.P. and C.B. Ching: Dynamics of simulated moving bed adsorptive separation processes. *Sep. Sci. Technol.*, **32**, 1993-2010, 1997.

[3] Ruthven, D.M. and C.B. Ching: Counter-current and simulated moving bed adsorption separation processes. *Chem. Eng. Sci.*, **44**, 1011-1038, 1989.

[4] Strube, J. and H. Schmidt-Traub: Dynamic simulation of simulated moving bed chromatographic processes. *Comp. Chem. Engng.*, **20**, S641-S646, 1996.

[5] Kaczmarski, K., M. Mazzotti, G. Storti and M. Morbidelli: Modeling Fixed-Bed Adsorption Columns through Orthogonal Collocation on Moving Finite Elements. *Comp. Chem. Engng.*, **21**, 641-660, 1997.

[6] Zhong, G. and G. Guiochon: Analytical solution for the linear ideal model of simulated moving bed chromatography. *Chem. Eng. Sci.*, **51**, 4307-4319, 1996.

[7] Van Deemter, J., F. Zuiderweg and A. Klinkenberg: Longitudinal Diffusion and resistance to mass transfer as causes of nonideality in chromatography. *Chem. Eng. Sci.*, **5**, 271-280, 1956.

[8] Lapidus, L. and N. Amundsen: Mathematics of adsorption in beds IV. The effect of longitudinal diffusion in ion exchange and chromatographic columns. *J. Phys. Chem*, **56**, 984-988, 1952.

[9] Dünnebier, G., I. Weirich and K.-U. Klatt: Computationally efficient dynamic modeling and simulation of simulated moving bed chromatographic processes with linear isotherms. *Chem. Eng. Sci.*, **53**, 2537-2546, 1998.

[10] Wendroff, B.: An analysis of front tracking for chromatography. *Acta Appl. Mathematicae*, **30**, 265-285, 1993.

[11] Dünnebier, G. and K.-U. Klatt: Modeling of chromatographic separation processes using nonlinear wave theory. Proc. of IFAC DYCOPS-5, Corfu, 521-526, 1993.

[12] Gu, T.: *Mathematical modeling and scale up of liquid chromatography*. Springer, New York, 1995.

IDENTIFICATION OF LABORATORY CHEMICAL REACTOR IN CLOSED-LOOP VIA YOULA-KUCERA PARAMETERISATION

S. Kozka, J. Mikles, F. Jelenciak, J. Dzivak,
Department of Process Control, Faculty of Chemical Technology, Slovak University of Technology, Radlinskeho 9, 812 37 Bratislava, Slovak Republic,
e-mail: kozka@chelin.chtf.stuba.sk

Abstract. This paper deals with a development of mathematical model of the laboratory continuos stirred tank reactor for decomposing exothermic reaction. This model can be used as an initial model for system identification e.g. identification method based on Youla-Kucera parameterisation and control design.

Introduction

The literature contains a large number of papers that discuss the design and control of chemical reactors. Textbooks such as [2] and [3] present the fundamentals, but the emphasis is primarily on steady-state aspects.

The study of the examples, by direct computer experimentation, has been shown to lead to a positive improvement in the understanding of the physical systems and confidence in the ability to deal with chemical rate processes. Quite simple models can often be shown to give quite realistic representations of process phenomena. The methods, described in [5], are applicable to a wide range of differing applications, including process identification, the analysis and design of experiments, process design and optimisation, process control and plant safety, all of which are essential aspects of modern chemical technology.

In this contribution, we propose a nonlinear model of continuous stirred tank reactor (CSTR). This model can be used for parameter estimation of the plant and controller design. A collection of recently developed feedback controller design methods founded on plant model identification are presented in [4] and [9].

Feedback control of chemical reactors is a problem, which is made difficult by the inherent nonlinear nature of the involved mechanism. One of the particular control problems, which were most commonly investigated, is the temperature control of an exothermic irreversible reaction in a cooled continuously stirred tank reactor. The reaction and the reactor considered in this paper are quite simple.

The paper is outlined as follows. In the next section of this contribution, we briefly describe the laboratory chemical reactor. Further, we discuss the modelling technique for nonlinear mathematical model of laboratory reactor. Then, the data obtained from nonlinear model with the experimental data is compared and the final section contains some concluding remarks.

Description of the reactor

The illustrative scheme of CSTR is shown in Fig. 1 [1]. In the reactor an exothermic reaction, dissociation of hydrogen peroxide to oxygen and water $2H_2O_2 \xrightarrow{K_2Cr_2O_7} 2H_2O + O_2$ is studied. As the catalysator of this reaction $K_2Cr_2O_7$ has been used. The function of catalysator is to accelerate the chemical reaction. The influence of $K_2Cr_2O_7$ is presented in [6]. The term "exothermic reaction" means, that during process the reaction heat is generated by the reaction.

The reactor, Fig. 1, (liquid volume approximately 0.95 l) consists of a glass tube closed by two gas-tight stainless steel lids. The glass coil 9 with heat transfer area $0.065m^2$ represents the cooling system of reactor. Water has been used as a coolant. Two peristaltic pumps 1, 2 meter both reactants (H_2O_2 and $K_2Cr_2O_7$) and feed reactor near the mixer 5. The reaction products are taken away by an overflow 10, which also provides constant liquid volume. Products are divided into gas 7 and liquid phases 8. The thermometer 6 protected from corrosion by polyethylene shield measures the temperature in the reactor. The temperatures of inlet 3 and outlet coolant 4 are also measured. The pneumatic valve continuously controls the coolant feed.

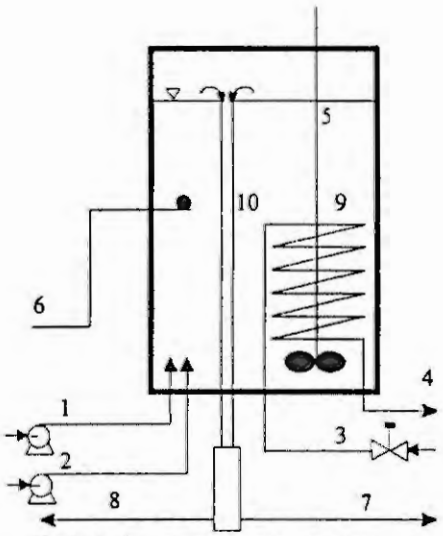


Fig.1: Laboratory reactor system

The nonlinear mathematical model of CSTR

Although CSTR normally operate at steady-state conditions, a development of the full dynamic equations for the system, is necessary to cover the instances of plant start up, shut down and the application of reactor control. Our mathematical model of laboratory CSTR consist of mass and energy balances respectively.

For steady-state operation of CSTR, there is no changes in conditions with the respect to time, and therefore the accumulation term is zero. Under transient conditions, the full form of the equation, involving all four terms, must be employed. In our case, the model has one mass balance of hydrogen peroxide in the following form

$$\frac{dc_A}{dt} = \frac{1}{V}\left(q_A c_{AV} - (q_A + q_B)c_A\right) - v(c_A, c_B, \vartheta) \qquad (1)$$

where

$$v(c_A, c_B, \vartheta) = 2kc_A^y c_B^z e^{\frac{E(\vartheta-\vartheta_0)}{R\vartheta\vartheta_0}}$$

$$c_B = \frac{c_{BV} q_B}{q_A + q_B}$$

Based on the law of conservation of energy, energy balance is a statement of the first law of thermodynamics. The internal energy depends, not only on temperature, but also on the mass of the system and its composition. For that reason, mass balance is almost always a necessary part of energy balancing.

In this case, the energy balance for the reactor can by written as

$$\frac{d\vartheta}{dt} = \frac{1}{C_P \rho}\frac{q_A + q_B}{V}(\vartheta_V - \vartheta) + (-\Delta H)v(c_A, c_B, \vartheta) - \frac{A\alpha}{C_P V \rho}(\vartheta - \vartheta_{CH}) - \frac{k_s A\alpha}{C_P V \rho}(\vartheta - \vartheta_{out}) \qquad (2)$$

Third equation of mathematical model of CSTR describes the heat transfer to and from reactor. Heat transfer is usually effected by coils or jackets, but can also be achieved by the use of external loop heat exchangers and, in certain cases, by the vaporisation of volatile material from the reactor. In this reactor the heat transfer is realised by the jacket.

In simple cases the jacket or cooling temperature may be assumed to be constant. In more complex dynamic problems, however, it may be necessary to allow for the dynamics of the cooling jacket, in which case temperature becomes a system variable. Under conditions, where the reactor and the jacket are well insulated and heat loss to the surroundings and mechanical work may be neglected we can write

$$\frac{d\vartheta_{CH}}{dt} = \frac{q_{CH}}{V_{CH}}(\vartheta_{CHV} - \vartheta_{CH}) + \frac{A\alpha}{C_{PCH} V_{CH} \rho_{CH}}(\vartheta - \vartheta_{CH}) \qquad (3)$$

Notation

C_A	H_2O_2 concentration [mol.m^{-3}]
C_{AV}	inlet H_2O_2 concentration [mol.m^{-3}]
C_B	$K_2Cr_2O_7$ concentration [mol.m^{-3}]
C_{BV}	inlet $K_2Cr_2O_7$ concentration [mol.m^{-3}]
q_A	H_2O_2 feed-rate [m^3.s^{-1}]
q_B	$K_2Cr_2O_7$ feed-rate [m^3.s^{-1}]
q_{CH}	coolant feed-rate [m^3.s^{-1}]
v	specific reaction rate [mol.m^{-3}.s^{-1}]
V	liquid volume of the reactor [m^3]
V_{CH}	volume of coolant in the reactor [m^3]
ρ	density of reactor contents [kg.m^{-3}]
ρ_{CH}	density of the coolant [kg.m^{-3}]
C_P	heat capacity of the reactor contents [J.kg^{-1}.K^{-1}]
C_{PCH}	heat capacity of the coolant [J.kg^{-1}.K^{-1}]
ϑ	reactor temperature [K]
ϑ_0	defined temperature for constant k, y and z [K]
ϑ_V	inlet temperature of reactant [K]
ϑ_{CH}	outlet temperature of the coolant [K]
ϑ_{CHV}	inlet temperature of the coolant [K]
ϑ_{OUT}	outer temperature [K]
A	heat transfer area [m^2]
α	heat transfer coefficient [W.m^{-2}.K^{-1}]
k	reaction rate constant [s^{-1}]
k_s	loss of heat coefficient [1]
y, z	the orders of the reaction [1]
R	gas constant [J.mol^{-1}. K^{-1}]
E	activation energy [J.mol^{-1}]
$-\Delta H$	heat of the reaction [J.mol^{-1}]

For the development of mathematics model of CSTR the following assumptions are made: neglected heat capacity of wall of the reactor and glass coil, constant density and specific heat capacity of liquid in the reactor, the constant overall heat transfer constants and constants $-\Delta H$, k_s, E. As the reactor is well mixed, the outlet stream concentrations and temperature are identical with those in the tank.

Results from the laboratory reactor

In this section, the presented nonlinear model of the continuous stirred tank reactor will be compared with real laboratory reactor in the situation where reactor temperature is the controlled output. The nominal reactor temperature from the safe steady-state point of view has been chosen as the lowest possible temperature. The value of this temperature is 31.8 degree of Celsius approximately.

Our aim is to compare the transient responses of the laboratory chemical reactor and its nonlinear model during transition from the nominal to the new steady-state. In this experiment, the new steady-state operating temperature of reactor was 35.8 degree of Celsius.

As shown in Fig. 2. for this experiment the nonlinear model provided similar results as the original represented by the laboratory reactor. As stated in the introduction this model of CSTR can be used as initial model for system identification based on Youla-Kucera parameterisation. The theoretical principles of this identification method can be found e.g. [7] and [8].

If the identification of laboratory chemical reactor based on Youla-Kucera parameterisation is used, we can obtain transfer function of reactor (around steady-state) in the following form

$$G(s) = \frac{b_0 + b_1 s}{1 + a_1 s + a_2 s^2} \qquad (4)$$

where s is operator of Laplace transformation and coefficients b_0, b_1, a_1, a_2 are functions of physical and chemical parameters of nonlinear model (1), (2), (3).

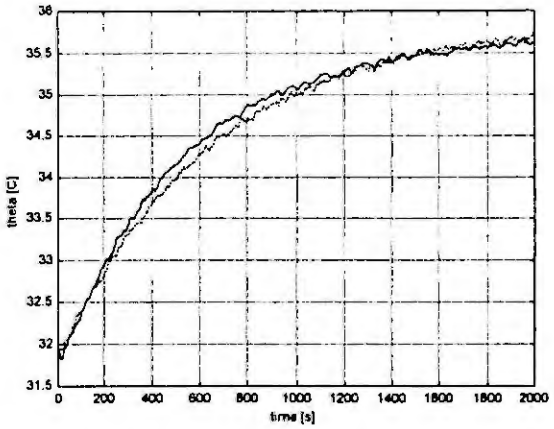


Fig. 2. Transient response of chemical reactor, experiment (solid), simulation (dotted).

Conclusions

This contribution examined the development of nonlinear mathematical model for laboratory continuous stirred tank reactor. The experimental data collected during experiment are compared with data from simulation. From Fig. 2 is clear, that the simple model can approximate the real process for given conditions.

The results show that presented model of chemical reactor can be successfully applied for closed-loop identification and control design respectively.

Acknowledgement: The work has been supported by VEGA MSSR (grants no. 1/5220/98, 1/4200/97 and PL 977010).

References

1. Dzivak, J. Mikles, J. Kozka, S. Jelenciak, F. and Dvoran, D., Model of chemical reactor for decomposition reaction, Selected Topics in Modeling and Control, Vol. 2, 1999, 87-89.
2. Floger, H. S., Elements of Chemical Reaction Engeneering, 2nd ed, Prentice Hall, Englewood Cliffs, NJ, 1992.
3. Froment, G. F. and Bischoff, K. B., Chemical Reactor Analysis and Design, 2nd ed, Wiley, New York, 1990.
4. Gevers, M., Towards a joint design of identification and control? In H.L. Trentelman J.C.Willems (Eds), Essays on Control: Perspective in the Theory and its Applications, Brikhauser, Boston, 1993, 111-151.
5. Ingham, J. Dunn, E. Heinzle, J. and Prenosil, E., Chemical Engeneering Dynamics,VCH, New York, 1994.
6. Kotocova, A. and Dugovicova, T., Kinetics of a Homogenous Decomposition of Hydrogen Peroxide, BC. thesis, ChTF STU Bratislava, 1997.
7. Mikles, J. Kozka, S. Meszaros, A. Fikar, M. Keseli, R., An iterative scheme for identification and control design with application to a chemical reactor, Proceedings of the 1st Workshop Polynomial Systems Theory and Applications, 15.-16.4.1999, Glasgow, 1999, 113-119.
8. Schrama, R.J.P., Approximate Identification and Control Design, PhD thesis, Delph University of Tchnology, The Netherlands, 1992.
9. Van den Hof, P.M.J. and Schrama, R.J.P., Identification and control – closed-loop issues, Automatica 31, 1995, 1751-1770.

DISTURBANCE REJECTION BY MEASUREMENT FEEDBACK FOR BOND GRAPH MODELS

J. Arib, C. Sueur, G. Dauphin-Tanguy.
L.A.I.L, U.P.R.E.S.A. C.N.R.S. 8021, Ecole Centrale de Lille,
B.P. 48, 59651 Villeneuve d'Ascq cedex, France

December 2, 1999

Abstract

In this paper the problem of disturbance rejection by measurement feedback (DRMF) is solved with the bond graph modelling. This tool gives the possibility to prove graphically if the DRMF problem is solvable or not, using the concept of causal paths. Thus it enables to place the sensors for measured outputs in suitable places in order to satisfy the condition for solvability and then obtain a solvable problem. Then, a control law for measurement feedback using a compensator is introduced. This control law is completely computed in bond graph models thanks to causal manipulations and allows pole assignment.

Introduction

The so-called disturbance rejection problem via measurement feedback has been extensively studied in the automatic control literature using several approaches such as geometric [1], [2] and [3] and structural one [4]and [5]. The purpose is to annul the influence of the disturbance inputs in the regulated outputs using a feedback loop with the measured outputs.

When some state variables are not available for measurement, disturbance rejection can not be achieved with static state feedback control law, that is why another technic is pointed out : measurement feedback. In this technic, only measured states are used in the feedback loop.

The computation of the control law for disturbance rejection problem by measurement feedback can be made with the geometric approach using the concept of (C, A, B)-pairs [1]. A (C, A, B)-pair is constituted of an $(A, \mathcal{B})$ invariant subspace and of a $(\mathcal{C}, A)$ invariant subspace. This concept allows the computation in two separated problems: the computation of a state feedback matrix and the computation of an output injection matrix. After that, the control law for measurement feedback is deduced. Then using these two invariant subspaces, a compensator is introduced in order to reject disturbance and to assign free modes. One part of the problem has already been solved [8] with the bond graph approach. The state feedback matrix is characterized using the supremal $(A, \mathcal{B})$ invariant subspace. The second one (dual notion of the first problem) is carried out using the infimal $(\mathcal{C}, A)$ invariant subspace which is easily determined with bond graph representation. In this paper, it is shown how the bond graph representation contributes in making the disturbance rejection problem easy, with only causal manipulations. First, some useful notations are recalled, then the formulation of the problem is highlighted using transfer matrix and geometric approach. In the second part, the geometric condition is given. Then, a bond graph method to compute the two sets of row and column essential orders is emphasized. These sets enable the determination of the structural necessary conditions or of the structural sufficient conditions for solvability.

A bond graph example is given to highlight the proposed methodologies which can be decomposed into several steps : first verify the structural conditions for solvability, second compute the supremal $(A, \mathcal{B})$ invariant subspace and the infimal $(\mathcal{C}, A)$ invariant subspace of the bond graph model, and third compute the full order compensator with the determination of the fixed and free modes.

Notations and problem formulation

Consider the linear time invariant systems (A, B, C, D, E) described by equation (1)

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) + Dh(t) \\ z(t) = Ex(t) \\ y(t) = Cx(t) \end{cases} \tag{1}$$

where $x(t) \in \mathcal{X} \approx \mathcal{R}^n$ is the state, $u(t) \in \mathcal{U} \approx \mathcal{R}^m$ is the control input, $h(t) \in \mathcal{H} \approx \mathcal{R}^q$ is the disturbance input, $z(t) \in \mathcal{Z} \approx \mathcal{R}^m$ is the output to be controlled and $y(t) \in \mathcal{Y} \approx \mathcal{R}^q$ is the measured output, and $A : \mathcal{X} \to \mathcal{X}$, $B : \mathcal{U} \to \mathcal{X}$, $C : \mathcal{X} \to \mathcal{Y}$, $D : \mathcal{H} \to \mathcal{X}$ and $E : \mathcal{X} \to \mathcal{Z}$. $\mathcal{B}$ is the image of B, $\mathcal{D}$ the image of D, $\mathcal{C}$ the kernel of C, $\mathcal{E}$ the kernel of E, B^j the j^{th} column of B and c_i is the i^{th} row of C.

Only square (right and left invertible) systems (E, A, B) and (C, A, D) are considered here, with A invertible. The Disturbance Rejection problem by Measurement Feedback is denoted as DRMF.

Consider the following measurement feedback processor [1]: $\begin{cases} \dot{w}(t) = Nw(t) + My(t) \\ u(t) = Lw(t) + Ky(t) \end{cases}$ where $w(t) \in \mathcal{W} \approx \mathcal{R}^w$ is the state of the compensator. The system (1) with the measurement feedback compensator in the extended state space $\mathcal{X}_w = \mathcal{X} \oplus \mathcal{W}$ is described by: $\begin{cases} \dot{x_w}(t) = A_w x_w(t) + D_w h(t) \\ z(t) = E_w x_w(t) \end{cases}$ where $A_w = \begin{bmatrix} A + BKC & BL \\ MC & N \end{bmatrix}$, $D_w = \begin{bmatrix} D \\ 0 \end{bmatrix}$, and $E_w = \begin{bmatrix} E & 0 \end{bmatrix}$.

DRMF Problem formulation: Find if possible a feedback compensator for (1) given such that in the closed loop system, the controlled outputs $z(t)$ does not depend on the disturbance inputs $h(t)$ i.e. $\begin{bmatrix} D & 0 \end{bmatrix} . (sI - A_w)^{-1} . \begin{bmatrix} E \\ 0 \end{bmatrix} = 0$

Geometric approach

In our approach, the geometric tools are used in order to give the conditions for solvability and to characterize the control law. It can be easily determined using the geometric subspaces like the $(A, \mathcal{B})$ invariant subspace or the $(\mathcal{C}, A)$ invariant subspace. In this part, necessary tools for the bond graph approach are recalled.

Solvability conditions for DRMF

Let us recall some geometric concepts. A subspace $\mathcal{V}$ of $\mathcal{X}$ is called $(A,\mathcal{B})$ invariant subspace if $A\mathcal{V} \subset \mathcal{V} + \mathcal{B}$, or if there exists a matrix $F \in \mathcal{F}(\mathcal{V})$ such that $(A + BF)\mathcal{V} \subset \mathcal{V}$. A subspace $\mathcal{S}$ of $\mathcal{X}$ is called $(\mathcal{C}, A)$ invariant subspace if $A(\mathcal{S} \cap \mathcal{C}) \subset \mathcal{S}$, or if there exists a matrix $G \in \mathcal{G}(\mathcal{S})$ such that $(A + GC)\mathcal{S} \subset \mathcal{S}$.

In this paper, the following geometric notations are used [6]: $\mathcal{V}^*_{(\mathcal{B},\mathcal{E})}$, the supremal $(A,\mathcal{B})$ invariant subspace contained in $\mathcal{E}$ is the limit of the non increasing algorithm (2) and $\mathcal{S}^*_{(\mathcal{C},\mathcal{D})}$, the infimal $(\mathcal{C}, A)$ invariant subspace containing $\mathcal{D}$, is the limit of the non decreasing algorithm (3).

$$\begin{cases} \mathcal{V}^0_{(\mathcal{B},\mathcal{E})} = \mathcal{X} \\ \mathcal{V}^{i+1}_{(\mathcal{B},\mathcal{E})} = \mathcal{E} \cap A^{-1}(\mathcal{V}^i_{(\mathcal{B},\mathcal{E})} + \mathcal{B}),\ i = 0, ..., n-1 \end{cases} \tag{2}$$

$$\begin{cases} \mathcal{S}^0_{(\mathcal{C},\mathcal{D})} = 0 \\ \mathcal{S}^{i+1}_{(\mathcal{C},\mathcal{D})} = A(\mathcal{S}^i_{(\mathcal{C},\mathcal{D})} \cap \mathcal{C}) + \mathcal{D},\ i = 0, ..., n-1 \end{cases} \tag{3}$$

Definition 0.1 *[1]A pair of subspace of $\mathcal{X}$, say $(\mathcal{S}, \mathcal{V})$, is called a $(\mathcal{C}, A, \mathcal{B})$-pair if:*

1. *$\mathcal{S}$ is a $(\mathcal{C}, A)$ invariant subspace.*
2. *$\mathcal{V}$ is an $(A,\mathcal{B})$ invariant subspace.*
3. *$\mathcal{S} \subset \mathcal{V}$.*

Schumacher [1] has proved that the DRMF problem can be formulated by using $(\mathcal{C}, A, \mathcal{B})$-pairs. Let us give the geometric solvability condition for the DRMF problem:

Theorem 0.2 *[2]The DRMF problem is solvable if and only if: $\mathcal{S}^*_{(\mathcal{C},\mathcal{D})} \subset \mathcal{V}^*_{(\mathcal{B},\mathcal{E})}$*

Control law for DRMF problem

After analysis, the control law consists of finding the full order feedback compensator.Using this kind of control law, it can be noticed that the fixed modes can be decomposed in two parts [3], the first one deals with the fixed modes computed with state feedback matrix F and the second one deals with the fixed modes computed with output injection matrix G [2].

Full order compensator for measurement feedback

The particular control law used in this section has some spectral characteristics, because it allows the free modes assignment [2] according to the following property:

Proposition 0.3 *[3]For a given $(\mathcal{S},\mathcal{V})$ a $(\mathcal{C},A,\mathcal{B})$-pair solution to the DRMF problem, there exists a set of poles of the closed loop system with the associated feedback compensator given by: $\sigma_{fix}(\mathcal{S},\mathcal{V}) = \sigma_{fix}(\mathcal{V}) \overset{\circ}{\cup} \sigma_{fix}(\mathcal{S})$*

Definition 0.4 *[2]**Observer-Based Full-Order Compensator.** Given the resolvent pair $(\mathcal{S},\mathcal{V})$, determine the matrices L_1, L_2, F, G such that:*

$$\begin{cases} N := A + GC + BFL_2 \\ M := BFL_1 - G \\ L := FL_2 \\ K := FL_1 \end{cases} \tag{4}$$

where $G \in \mathcal{G}(\mathcal{S})$ is an output injection matrix, and $F \in \mathcal{F}(\mathcal{V})$ is a state feedback matrix, and L_1, L_2 are such that

$$\begin{cases} L_1C + L_2 = I_{n*n} \\ \ker L_2 \oplus (\mathcal{S} \cap \mathcal{C}) = \mathcal{S}. \end{cases} \tag{5}$$

After the control law determination, let us now determine the fixed modes of DRMF problem.

Proposition 0.5 *Given a system (A,B,C,D,E) described by (1), and given $(\mathcal{S}^*_{(\mathcal{C},\mathcal{D})},\mathcal{V}^*_{(\mathcal{B},\mathcal{E})})$ a $(\mathcal{C},A,\mathcal{B})$-pair. Then the fixed modes of DRMF problem in closed loop are given by $\sigma_{fix}(\mathcal{S}^*_{(\mathcal{C},\mathcal{D})},\mathcal{V}^*_{(\mathcal{B},\mathcal{E})}) = \sigma_{fix}(\mathcal{V}^*_{(\mathcal{B},\mathcal{E})}) \overset{\circ}{\cup} \sigma_{fix}(\mathcal{S}^*_{(\mathcal{C},\mathcal{D})})\sigma_{fix}(\mathcal{V}^*_{(\mathcal{B},\mathcal{E})})$ = invariant zeros of (E,A,B) [8] and $\sigma_{fix}(\mathcal{S}^*_{(\mathcal{C},\mathcal{D})})$ = invariant zeros of (C,A,D).*

Bond graph approach

In this paper, the bond graph approach is emphasized for (A,B) controllable and (C,A) observable models. From the bond graph representation, it is possible to point out graphically the solvability of the DRMF problem. First, some basic concepts, needed in the disturbance rejection problem by output feedback are recalled.

The disturbance rejection problem can be solved with the cansal path concept on a bond graph model [7]. Some results about infinite zero order for bond graph models are recalled [7]. It is supposed that each dynamical element has an integral causality assignment if the preferential integral causality assignment is chosen.

In this paper, the following structural elements are needed: $\{n'_i\}$ the global infinite structure, $\{n_i\}$ the row infinite structure and $\{n^c_j\}$ the column infinite structure [8].

Proposition 0.6 *h_{ij}, the $(i,j)^{th}$ infinite structure of a system (c_i,A,B^j), is equal to the length of the shortest causal path between the i^{th} output detector and the j^{th} input source.*

Now, let us recall a structural method to determine the row essential orders. Consider a system (C,A,B) and a subsystem $(\bar{C}_i,A,B)$ obtained by removing the i^{th} output detector from the model of system (C,A,B).

Let $\{n'_i\}$ be the global infinite structure of the system (C,A,B) and $\{n'^r_{ik}\}$ be the global infinite structure of the subsystem $(\bar{C}_k,A,B)$.

Proposition 0.7 *The i^{th} row essential order denoted n_{ie} verifies: $n_{ie} = \sum_k n'_k - \sum_k n'^r_{ik}$.*

The column essential orders for a given system (C, A, B) is obtained with a dual formulation. Consider a subsystem $(C, A, \bar{B}^j)$ obtained by removing the j^{th} input source from the bond graph model of (C, A, B) then:

Let $\{n'^c_{jk}\}$ be the global infinite structure of the subsystem $(C, A, \bar{B}^j)$.

Proposition 0.8 *The j^{th} column essential order denoted n^c_{je} verifies: $n^c_{je} = \sum_k n'_k - \sum_k n'^c_{jk}$.*

These structural sets give us the possibility to determine graphically the conditions for solvability of the disturbance problem by measurement feedback.

Structural necessary condition and sufficient condition [4]

In this section, first a necessary condition, then a sufficient condition for solvability of DRMF are recalled.

Theorem 0.9 *[4] Consider the system described by (1), the measurement feedback disturbance rejection has a solution only if*

$$h_{ij} \geq g_i + n^c_j \text{ for } i = 1, ..., m,\ j = 1, ..., q$$

where h_{ij} is the infinite zero order of the (i, j) entry $H_{ij}(s)$ i.e. the subsystem (E_i, A, D_j); g_i is the infinite zero order of the i^{th} row of $G(s)$ i.e. the subsystem (E_i, A, B); n^c_j is the infinite zero order of the j^{th} column of $Q(s)$ i.e. the subsystem (C, A, D^j).

A sufficient condition is recalled here.

Theorem 0.10 *[4] Consider the system described by (1), the output feedback disturbance rejection has a solution if*

$$h_{ij} \geq g_{ie} + n^c_{je} \text{ for } i = 1, ..., m,\ j = 1, ..., q$$

where h_{ij} is the infinite zero order of the (i, j) entry $H_{ij}(s)$ i.e. the subsystem (E_i, A, D^j); g_{ie} is the i^{th} row essential order of $G(s)$ i.e. the subsystem (E_i, A, B); n^c_{je} is the j^{th} column essential order of $Q(s)$ i.e. the subsystem (C, A, D^j).

The following theorem gives the condition to obtained necessary and sufficient conditions:

Theorem 0.11 *The necessary condition and sufficient condition turn out to be necessary and sufficient conditions if and only if*

$$n^c_j = n^c_{je};\ j = 1, ..., q.$$

In the following section, the control law is deduced in bond graph models.

Control law in bond graph

In the sequel, it is shown how bond graph modelling contributes to solve and to determine the control law for DRMF, using only causal path concept. The invariant subspaces $\mathcal{S}^*_{(C,\mathcal{D})}$ and the othogonal complement $\mathcal{V}^{*\perp}_{(\mathcal{B},\mathcal{E})}$ are easily characterized in bond graph models, and the control laws are deduced. The determination of the invariant subspaces $\mathcal{V}^*_{(\mathcal{B},\mathcal{E})}$ and $\mathcal{S}^*_{(C,\mathcal{D})}$ in bond graph model allows the calculation of the feedback matrix F and the measurement injection matrix G.

The state feedback matrix F is determined with bond graph methodology according to [8]:

$$[(\mathcal{V}^{*\perp}_{(\mathcal{B},\mathcal{E})})^t.B].F = -(\mathcal{V}^{*\perp}_{(\mathcal{B},\mathcal{E})})^t.A + LC(\mathcal{V}^{*\perp}_{(\mathcal{B},\mathcal{E})})^t. \quad (6)$$

In the same way, the output injection matrix G, is determined thanks to bond graph characterization of $\mathcal{S}^*_{(C,\mathcal{D})}$: $G[C.\mathcal{S}^*_{(C,\mathcal{D})}] = -A\mathcal{S}^*_{(C,\mathcal{D})} + LC(\mathcal{S}^*_{(C,\mathcal{D})})$, where LC means the linear combination of vectors forming the subspace.

Thanks to these matrices and to invariant subspaces, the full order compensator is deduced.

Remark 1 *The symbolic expression of $\mathcal{S}^*_{(C,\mathcal{D}^j)}$ for the j^{th} disturbance input is given:*

$$\mathcal{S}^*_{(C,\mathcal{D}_j)} = vect\{\mathcal{D}^j, A\mathcal{D}^j, ..., A^{n^c_j}\mathcal{D}^j\}.$$

For the DRMF using a full order compensator, only the two matrices F and G are needed. They can he pointed out from bond graph models as seen previously.

Example

Consider the bond graph model of figure 1 in integral causality assignment. The state vector is $x = [p_{I_1}, p_{I_2}, p_{I_3}, p_{I_4}, q_{C_1}, q_{C_2}, q_{C_3}, q_{C_4}, q_{C_5}]^t$. This model has 2 control inputs $Mse : u1$ and $Msf : u2$ and 2 controlled outputs $df : z1$ and $df : z2$ and 1 disturbance input $Se : h$ and 1 measured output det : $y1$.

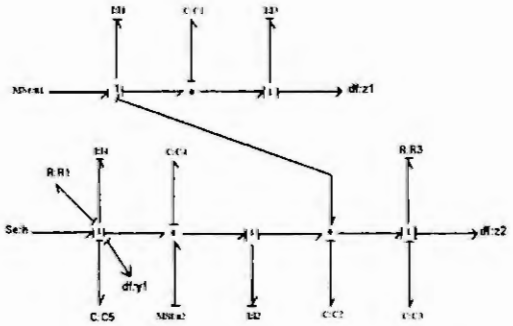


figure 1: Bond graph in integral causality

The necessary condition for solvability of DRMF problem is: $h_{ij} \geq g_i + n_j^c$ for $i = 1, 2$, $j = 1$.

Let us determine the integers h_{ij}. For this objective, the subsystems (E_i, A, D_1) are considered, then $h_{11} = 7$ and $h_{21} = 4$. The two corresponding causal paths are: $df : z1 - I3 - C1 - I1 - C2 - I2 - C4 - I4 - Se : h$ and $df : z2 - R3 - C2 - I2 - C4 - I4 - Se : h$. For the integers g_i, the subsystems (E_i, A, B) are considered, then $g_1 = 3$ and $g_2 = 2$. The two corresponding causal paths are: $df : z1 - I3 - C1 - I1 - MSe : u1$ and $df : z2 - R3 - C2 - I1 - Mse : u1$. And at last, for the integer n_1^c, the subsystem (C, A, D_1) is considered, then $n_1^c = 1$. The corresponding causal path is: $df : y1 - I4 - Se : h$. Thus, for $i = 1, 2$, $j = 1$, the condition $h_{ij} \geq g_i + n_j^c$ is verified and it is a necessary and sufficient condition because $n_1^c = n_{1e}^c$, then the DRMF problem is solvable.

As it has known, thanks to the separation property, the computation of tbe control law for DRMF problem can be divided in two problems: the computation of state feedback matrix and the computation of output injection matrix.

Computation of state feedback matrix F

The state feedback matrix is computed by considering only the system (E, A, B). The global infinite structure of the system $[G(s)]$ is $\{3, 3\}$ because the two different and shortest input-output causal paths are: $df : z1 - I3 - C1 - I1 - MSe : u1$ and $df : z2 - R3 - C2 - I2 - C4 - MSf : u2$ [8]. The global infinite structure of $[s^{-1}G(s)]$ is obtained by adding a causal path of length one to control inputs, thus it is equal to $\{4, 4\}$. In [8], it is seen that $\dim(\mathcal{V}^{*\perp}_{(\mathcal{B},\mathcal{E})}) = \sum_{i=1}^{i=2} n_i' = 6$, the bond graph modelling allows the formal computation of $\mathcal{V}^{*\perp}_{(\mathcal{B},\mathcal{E})}$:

$$\mathcal{V}^{*\perp}_{(\mathcal{B},\mathcal{E})} = vect\{(E_1)^t, (E_1A)^t, (E_1A^2)^t, (E_2)^t, (E_2A)^t, [I1C2R3.(E_2A^2)^t - I3C1I1.(E_1A^3)^t]\}.$$

Using this formal expression, the control law is deduced from equation (6), and it contains 12 parameters for pole assignment.

After applying this control law with $\mathcal{V}^{*}_{(\mathcal{B},\mathcal{E})}$ as geometric support, the fixed modes become the invariant zeros: $\sigma_{fix}(\mathcal{V}^{*}_{(\mathcal{B},\mathcal{E})}) = invariant\ zeros\ of\ (E, A, B)$

The control law allows the introduction of 6 degrees of freedom because of $\dim(\mathcal{V}^{*\perp}_{(\mathcal{B},\mathcal{E})}) = 6$, then there exists 3 fixed modes which are at the same time invariant zeros: 2 stable invariant zeros: $(-C5 + \sqrt{C5^2 - 4I4C5R1^2})/I4C5R1$ and $(-C5 - \sqrt{C5^2 - 4I4C5R1^2})/I4C5R1$ and 1 null invariant zero (instable) [8]. These invariant zeros are obtained with the Smith form of the matrix system.

Computation of output injection matrix G

In order to compute the output injection matrix G, consider only the system (C, A, D), the invariant subspace $S^{*}_{(C,\mathcal{D})}$ must be determined. For this, consider the column infinite structure of the system $(C, A, D) : n_j^c = 1$

and thanks to causal path the expression (7) is obtained:

$$\mathcal{S}^*_{(C,\mathcal{D})} = \mathcal{D} = vect\{(0,0,0,1,0,0,0,0,0)^t\}. \quad (7)$$

Then according to (8):

$$G.[C\mathcal{D}] = -A\mathcal{D} + LC(\mathcal{D}). \quad (8)$$

where LC means the linear combination.

After applying this control law with $\mathcal{S}^*_{(C,\mathcal{D})}$ as geometric support, the fixed modes become the invariant zeros: $\sigma_{fix}(\mathcal{S}^*_{(C,\mathcal{D})}) = invariant\ zeros\ of\ (C,A,D)$

The control law allows the introduction of 1 degree of freedom because of $\dim(\mathcal{S}^*_{(C,\mathcal{D})}) = 1$, then there exists 9 fixed modes which are at the same time invariant zeros: 8 stable invariant zeros and 1 null invariant zero (instable).

Full order compensator for measurement feedback

In order to do pole assignment, the control law is introduced, using a feedback observation. This compensator seen in definition 0.4 uses the matrices F and G already determined in bond graph models. The bond graph methodology is used here because it allows us to determine the symbolic expression of matrices constituting the compensator. Then the matrices K and N and M and L are calculated formally thanks to (4) and (5).

In this example, the order of the extended system is $n + \dim(\mathcal{W}) = 18$. Using this control law based on $(\mathcal{S}^*_{(C,\mathcal{D})}, \mathcal{V}^*_{(\mathcal{B},\mathcal{E})})$ as a resolvent $(C,A,\mathcal{B})$-pair, the fixed modes become the union of fixed modes of $\mathcal{V}^*_{(\mathcal{B},\mathcal{E})}$ and $\mathcal{S}^*_{(C,\mathcal{D})}$, then the system has 12 fixed modes: 10 stable invariant zeros and 2 null invariant zeros (instable).

Conclusion

In this paper, the DRMF problem is highlighted in bond graph models, it has shown that bond graph modelling contributes easily to study the solvability of this problem and to make it solvable with choosing the right position for detectors. For sake of place, the problem of DRMF with stability is emphasized in an another article where new conditions for stability are proved and the control law which insnres stability is given with the bond graph approach.

References

[1] Schumacher J.M. Compensator Synthesis Using (C,A,B)-pairs. *IEEE Transactions on Automatic Control*, Vol, AC-25, No. 6, pp 1133-1138, December, (1980).

[2] Basile, G and Marro G. Controlled invariants and conditioned invariants in linear system theory. *Englewood Cliffs, Prentice Hall*, New Jersey, USA, (1992).

[3] Del-Murro-Cuellar B., and M. Malabre. Fixed Poles of Disturbance Rejection by Dynamic Output Feedback: Geometric and Structural Approaches. 4^{th} *European Control Conference ECC'97*, Brussels, Belgium, (1997).

[4] Commault C., J.M. Dion and Benhacen M. Output Feedback Disturbance Decoupling Graph Interpretation for Structured Systems. *Automatica*, Vol 29, No.6, pp 1463-1472, (1993).

[5] Van Der Woude J.W. Graph Theoretic Conditions for Structural Disturbance Decoupling With Pole placement.

[6] Wonham W.M. Geometric State-Space Theory in Linear Multivariable Control : *a status report. Automatica*, Vol. 15, pp 5-13, (1979).

[7] Bertrand, J.M.,Sueur C. and Dauphin-Tanguy G. On the finite and infinite structures of bond-graph models. *IEEE Conference on Systems, Man and Cybernetics*, Orlando, Florida, USA, pp. 2472-2477, (1997).

[8] Arib J, Sueur C. and Dauphin-Tanguy G. Disturbance Rejection with Stability for Bond Graph Models. Accepted in *European Control Conference ECC'99*, Karlsruhe, Germany (September 1999).

[9] Commault, C. and Dion, J.M. Structure at Infinity of Linear Multivariable Systems. A Geometric Approach. *IEEE Transactions on Automatic Control*, pp 693-696, (1982).

Describing bond graph models of hydraulic components in Modelica

W. Borutzky
Department of Electrical and Mechanical Engineering
Rhein-Sieg University of Applied Sciences, D-53754 Sankt Augustin, Germany

B. Barnard
Department of Mechanical Engineering
Monash University, Caulfield Campus, Victoria 3145, Australia

J. U. Thoma
Adjunct Professor, Department of Systems Design, University of Waterloo, Ontario, Canada and
Thoma Consulting, Bellevueweg 23, CH 6300 Zug, Switzerland

Abstract

In this paper we discuss an object oriented description of bond graph models of hydraulic components by means of the unified modelling language *Modelica*. A library which is still under development is briefly described and models of some standard hydraulic components are given for illustration. In particular we address the modelling of hydraulic orifices.
Keywords: Hydraulic components, orifices, bond graph models, unified modelling language, object oriented modelling, model exchange.

1 Introduction

For modelling and simulation of hydraulic systems a number of special purpose simulators (proprietary simulation software) along with their comprehensive model libraries can be used. Moreover, for some general purpose simulators like EASY5, Saber, or Dymola hydraulic libraries are available. Commonly hydraulic systems are described as a network or as a block diagram. In the first case component models are interconnected according to the physical structure of the system by applying Kirchhoff's current law generalized to volume flow rates. For a graphical representation of hydraulic circuits standard symbols are used. On the other hand bond graphs are well suited for modelling multi energy domain systems and have been used for a long time especially for modelling hydraulic systems [1], [5], [8]. In both worlds considerable engineering knowledge has been accumulated in model libraries. Unfortunately, since the user interface of simulators has been designed to support either generalized networks or bond graphs, models developed in one world can not be easily used in the other world. With the advent of the new modelling language *Modelica*[6] the situation has changed. Modelica has been developed in an international effort aiming at combining the concepts of various present object oriented modelling languages. A major goal of its design has been to promote the exchange and reuse of models between different (proprietary) simulator packages and to introduce in that way a new de facto standard. Concerning the modelling of energy flows Modelica also relies on the concept of generalized networks, but due to a small modification performed in the course of the design of the language, it may be used for the description of bond graph models as well. A description of the basic bond graph elements and of their interconnection in Modelica is rather straightforward [4], [3]. Moreover, bond graph modelling can be viewed as a special form of an object oriented modelling approach [3], [2]. On the other hand a Dymola model library for hydraulic components for connection to hydraulic circuits has been developed by Beater and a translation to Modelica has been announced [6]. In this paper we do not pursue the network based modelling approach. Inspired by the fact that bond graph models in principle can be described in the object oriented modelling language Modelica, and by the fact that this language is going to become a widely accepted neutral exchange format, this paper offers a small a library of bond graph models of hydraulic components described in Modelica. Since bond graphs have been used for a long time for describing hydraulic systems, the idea of a bond graph library for hydraulic components is not new. To our knowledge a Modelica description

of bond graph models of hydraulic components however has not been undertaken so far. The advantage is that, depending on what is more suitable in regard to the design task, either a circuit or a bond graph model of a hydraulic (sub)system can be composed from library models and processed by a modelling and simulation package like Dymola if the software supports the new language either directly or by means of a translator. On the other hand since future versions of bond graph modelling tools are expected to support Modelica, our hydraulic component library under development will be of general use and not just another add-on to a particular program.

2 Hydraulic component bond graph models in Modelica

Once the basic bond graph elements have been described in Modelica and stored in a bond graph library, it is an obvious step to use them for the description of bond graph models of standard hydraulic components and to store those descriptions in a hydraulic library. A straightforward way of implementing such a library is to keep the Modelica description of bond graph models of hydraulic components in a file that is added to a high level model of a hydraulic system containing essentially only a description of the connectivity between component models. Normally a number of models of different complexity are provided for standard components so accounting for different aspects. This is a way to make models as accurate as needed in a context under consideration and to keep them as simple as possible at the same time. If the library is just a file or a set of several files containing the Modelica code, good documentation is required in order to enable the designer to select an appropriate model for the actual design task. The next step towards better user support would be to provide a librarian procedure which accepts functional model specifications and allows for navigation through the library to find a model that meets given requirements. So far our implementation is just a collection of Modelica code for a growing number of bond graph models for standard components. Modelica's ability to define model classes and to support inheritance enables users to develop a suite of models for a component by starting from a rather simple model then adding other features. To illustrate how the models in the hydraulic component library are built, consider a small example. A constant displacement pump where the transformation of the delivered energy into hydraulic energy is not of concern, the pump may be represented by a modulated flow source and a hydraulic resistor accounting for internal leakage as shown in Fig. 1. The little square indicates the

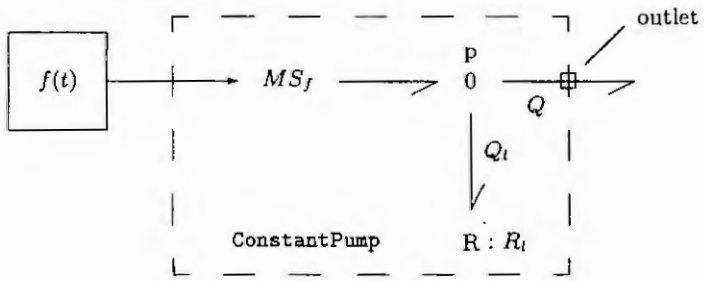


Figure 1: Simple bond graph model of a constant displacement pump

power port of the pump model labeled *outlet*. The time history of the volume flow rate, Q delivered by the pump at the hydraulic outlet port may be determined by connecting a corresponding functional block to the signal input port of the pump model. In Fig. 1 $f(t)$ may be a constant, a sin function or a table. A Modelica description of this simple model is given in Fig. 2. In the list of submodels the entry in the first column denotes the model class of which the submodel is an instance of while the local name of the submodel is given in the second column. In the equation section each connect statement corresponds to a bond. The simple model of Fig. 1 can be easily adapted to account for the transformation of mechanical into hydraulic power if the flow source is replaced by a power conserving two-port transformer bond graph element turning the component model into one with two power ports. Moreover, compressibility of the fluid in the outlet port may be accounted for by attaching a C-element to the 0-junction. By adding a signal input port and by replacing the transformer by a displacement modulated transformer (MTF), we obtain a bond graph model of a variable displacement pump controlled by the angle of inclination of the swashplate. A Modelica description of the bond graph model of this variable displacement pump is a simple extension of that in Fig. 2.

```
model ConstantPump

/* This bond graph model has
   - one power port labeled outlet which provides the volume flow rate Q,
   - a signal input denoted by u controlling the magnitude of the modulated source MSf,
   - one parameter accounting for losses due to internal leakage
   The time behavior of the flow delivered by the pump may be specified by
   connecting a corresponding signal generator block to the signal input.
*/
   PowerPort outlet ;
   input u;
   parameter Real Rleak (unit="Pa s/m3 ") = .2e12 ; // default value for pump leakage

/* Type and local name of submodels used */
   MSf       source;
   zero3P    p ;              // 0-junction represents the load pressure
   linR      Rl (R = Rleak);  // accounting for losses due to internal leakage

/* connectivity of power ports according the bond graph */
equation
   source.in = u;
   connect(source.outlet, p.port1);
   connect(p.port2, Rl.inlet);
   connect(p.port3, outlet);
end ConstantPump;
```

Figure 2: Modelica description of a bond graph model of a constant displacement pump

Having modelled a hydraulic pump as a power transducer the next step is to set up shaft models. The most simple model may be just a bond - that is, any losses and energy storage in the transmission are neglected.

3 Modelling hydraulic orifices

Hydraulic orifices are frequently described by the well known square root law

$$Q = c_d \cdot A \cdot \sqrt{\frac{2}{\rho} |\Delta p|} \cdot sign(\Delta p) \tag{1}$$

derived from Bernoulli's energy equation for an incompressible steady state flow. While A denotes the cross section of the restriction, the coefficient c_d accounts for energy losses. It depends on the geometry of the restriction and of the Reynolds number Re which characterizes the mode of the flow. Often a constant value holding for *turbulent* conditions is adopted. However, it is known that the discharge coefficient c_d is a non-linear function of $\sqrt{Re}$ [7]. If the so-called hydraulic diameter D_h of the orifice is known, the Reynolds number can be expressed by the volume flow rate Q.

$$Re = \frac{D_h}{A \cdot \nu} \cdot Q \tag{2}$$

In equation (2) the kinematic viscosity ν depends on the temperature and on the pressure. Often an average value is used. Observing that c_d is a function of Re by combining equations (1) and (2) we see that the volume flow rate through an orifice is given by an *implicit* non-linear equation of the form

$$Q = f(Q) \cdot A \cdot \sqrt{\frac{2}{\rho} |\Delta p|} \cdot sign(\Delta p) \tag{3}$$

A calculation of the volume flow rate based on such an equation is costly in regard to computational time since iteration is required. For small Reynolds numbers the relation between the discharge coefficient and the Reynolds number may be approximated by

$$c_d = k \cdot \sqrt{Re} \tag{4}$$

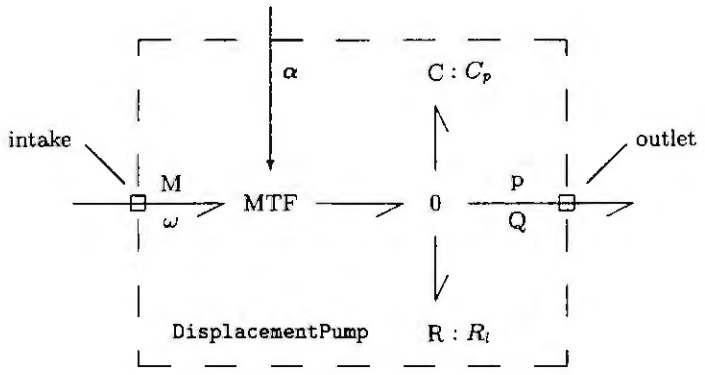


Figure 3: Bond graph model of a variable displacement pump

in which k is a constant. By substituting (4) into (2) and the result into (1) we obtain a *linear* relation between the volume flow rate Q and the pressure drop Δp across the orifice, for laminar flow.

$$Q = k^2 \frac{2A \cdot D_h}{\rho \cdot \nu} \cdot \Delta p \tag{5}$$

Consequently assuming c_d a constant for the sake of simplicity, there is not only a deviation from the linear relation for small pressure drops. Moreover, for small pressure drops the slope of the square root characteristic is very steep tending to infinity as $\Delta p \rightarrow 0$ which may cause problems for the numerical integration.

One possible remedy to this problem, and to more accurately encompass a larger range of flows, is to use a linear characteristic for Reynolds numbers below a critical value Re_{crit} and the square root characteristic or even a constant for $Re > Re_{crit}$. The problem with this obvious approach is that the value of Re_{crit} can only be estimated and that there is no unique tangent point, where this relation changes from one equation to the other. An alternative we are using for our orifice model is to approximate the non-linear relation $c_d = f_1(\sqrt{Re})$ by a simple relation for all Reynolds numbers.

$$c_d = \frac{c_{dmax} \cdot \sqrt{Re}}{\sqrt{Re} + \sqrt{Re_{crit}}} \tag{6}$$

For small Re values this approximation reduces to $c_d \approx c_{dmax}/\sqrt{Re_{crit}} \cdot \sqrt{Re}$. For large Reynolds numbers we have $c_d \approx c_{dmax} := 0.61$. Substituting equation (2) into (6) and substituting the result into equation (1) yields a quadratic equation for $\sqrt{|Q|}$ instead of an implicit non-linear relation for Q of the form given by equation (3).

$$\sqrt{\frac{D_h}{A \cdot \nu}} \cdot (\sqrt{|Q|})^2 + \sqrt{Re_{crit}} \cdot \sqrt{|Q|} - c_{dmax} \cdot \sqrt{\frac{D_h}{A \cdot \nu}} \cdot A \cdot \sqrt{\frac{2}{\rho} |\Delta p|} = 0 \tag{7}$$

This quadratic equation for $\sqrt{|Q|}$ has one unique solution. This solution, defined as z, is used in the Modelica description of the bond graph model of an orifice as given in Fig. 4. The model class `orifice` has been derived from the superclass `passiveOnePort`. More precisely it can be seen as a specialization of a class `OnePortResistor` which is specified by the requirements that the relation between the port variables is an algebraic one and that their product is positive at any time instant. However, such general properties are hard to express in a modeling language. The model `Orifice` can be easily modified so that it describes an orifice in a spool valve with a cross section area that varies with the spool displacement. We just need to add a signal port for the displacement and a formula for the cross section area. The result can be considered a specialization of a general class `modulatedOnePortResistor`.

4 Conclusion

Inspired by the fact that bond graph elements can be described in Modelica straightforwardly by exploiting the object oriented features of the language and by the fact that Modelica is going to become a widely

```
model Orifice "hydraulic orifice"

/* This model describes a steady state fluid flow through an orifice of fixed cross section Area.
   The non-linear constitutive law for the discharge coefficient is approximated by:

      cd(Re) = cdmax * sqrt(Re) / (sqrt(Re) + sqrt(Recrit))

   This approximation holds for turbulent as well as laminar flow.
   The possibility of cavitation is not taken into account.
*/
   extends passiveOnePort ; // makes power variables p.e and p.f of port p available
// inherit constants describing fluid properties like density rho and kinematic viscosity nu
   extends Modelica.Constant ;
   parameter Real Area ; // cross section area
   parameter Real Dh ; // hydraulic diameter
   parameter Real cdmax = 0.611 ;
   parameter Real Recrit = 9.33 ;
   parameter Real K = sqrt(Dh / Area / nu) ;
   parameter Real k1 = sqrt(Recrit)/2/K ;
   parameter Real k2 = cdmax * Area * sqrt(2/rho) ;
   Real Q    "volume flow rate through the orifice" ;
   Real dp   "pressure drop across the orifice" ;
   Real z    " auxiliary variable" ;
equation
   Q  =  p.f ;
   dp =  p.e ;
   z  =  sqrt(k1^2 + k2 * sqrt(abs(dp))) - k1 ;
   Q  =  z^2 * sign(dp) ;
end Orifice
```

Figure 4: Bond graph model of an orifice in Modelica

accepted neutral exchange format promoting the exchange and reuse of models we have been using Modelica to describe bond graph models of hydraulic components. The advantage is that the language Modelica is not bound to a specific (proprietary) modeling tool. Either it can be understood directly or processed by an import facility. In any case, Modelica although not primarily designed for supporting bond graph modeling may help promote the exchange of bond graph models. A particular issue in this paper has been an appropriate model of hydraulic orifices accounting for the change from turbulent to laminar flow conditions while the pressure drop across the orifice tends to small values.

References

[1] B. W. Barnard. Predicting the dynamic response of a hydraulic system using power bond graphs. Master's thesis, Monash University, Melbourne, Australia, 1973.

[2] W. Borutzky. Bond graph modeling from an object oriented modeling point of view. *Simulation Practice and Theory*, 1999.

[3] W. Borutzky. Relations between bond graph and object-oriented physical systems modeling. In J. J. Granda and F. E. Cellier, editors, *ICBGM'99, 4th International Conference on Bond Graph Modeling and Simulation*, pages 11–17. SCS Publishing, 1999. Simulation Series, Vol. 31, No. 1, ISBN: 1-56555-155-9.

[4] J. F. Broenink. Bond-graph modeling in Modelica. In W. Hahn, A. Lehmann, W. Borutzky, and H. Ziegler, editors, *Simulation in Industry, 9th European Simulation Symposium 1997, ESS'97*, pages 137–141, 1997. Passau, Germany.

[5] P. Dransfield. *Hydraulic Control Systems - Design and Analysis of Their Dynamics.* Springer-Verlag, New York, 1981.

[6] H. Elmqvist et. al. *Modelica^{TM} – A Unified Object-Oriented Language for Physical Systems Modeling Version 1*, Sept. 1997. http://www.Modelica.org.

[7] H. E. Merritt. *Hydraulic Control Systems.* Wiley & Sons, 1967.

[8] J. U. Thoma. *Simulation by Bondgraphs - Introduction to a Graphical Method.* Springer-Verlag, New York, 1990.

MODELLING AND SIMULATION OF HYDRAULIC STEPPER CYLINDER BY BOND GRAPH METHOD

R. Dindorf
Kielce University of Technology
Al. Tysiaclecia Panstwa Polskiego 7, PL–25-314 Kielce, Poland

Abstract. The present paper describes dynamic modelling and results of simulation investigation of a hydraulic stepper cylinder. The method of bond graph is used in modelling dynamics of the considered cylinder design. The dynamic model of a hydraulic stepper cylinder has been extended by conduits and supply pipeline. For this purpose a new element of bond graph defined as double bond – DB is suggested. The application of bond graphs with a new element DB in modelling of pulsating flow in supply pipelines is described.

Introduction

Hydraulic stepper cylinders make it possible to obtain some precisely definite positions of piston. These cylinders find application in moving ports of industrial manipulators and robots or they can perform auxiliary functions in the drives of technological machines [2.]. Stepper cylinders have to satisfy such requirements, as high reliability and speed, precision of performance, irrespective of the properties of working fluid and the rate of piston movement. The piston moves in the desired direction to reach the point precisely opposite this port. In steady state pressure on both sides of the piston is equalized, and fluid flows through uniform holes to the tank. In order to secure proper dynamic model of the stepper cylinder conduits and pipeline is introduced. Bond graphs [4] are used for modelling of the dynamics of the stepper cylinder. In selecting a method of dynamics modelling it is taken into consideration that the dynamic structure of bond graphs is closely identifiable with the functional structure of the hydraulic system [3].

Bond graph of dynamic model of stepper cylinder

Schematic representation of a hydraulic stepper cylinder is presented in Fig.1a. In the adopted design of a hydraulic stepper cylinder, presented in a simplified way in Fig.1b, four positions of the piston are obtained. The stepper cylinder is fed from the source of constant pressure. In the inlets to the right and left chambers of the cylinder throttle valves (the capillary and orifice type) of constant diameter are fixed. In the cylinder a sleeve with properly made grooves and holes is fixed. Turning the sleeve by angles 0 and 180 one obtains positions 1 and 3. Turning it by angles 90 and 270 positions 2 and 4 are obtained. In each position of the piston the cylinder is connected with the return line to the tank through four holes. The piston assumes left position 1 or 2 when control shutoff valve I is open and control shutoff valve II is closed, and it assumes right position 3 or 4 when the control shutoff valve I is closed and control shutoff valve II is open. For example, opening of the control shutoff valve I and closing of the shutoff valve II cause decrease of pressure p_1 and quick displacement of the piston from the right to the left position. As the result of volume increase in chamber 1 and volume decrease in chamber 2 pressure p_1 increases and pressure p_2 decreases. The pressure in the left and right chambers of the cylinder increases and decreases alternately until the steady position of the piston is achieved. The piston position is established after achieving equilibrium state of forces acting on the cylinder piston.

The method of bond graphs is used for modelling of dynamics of a stepper cylinder. In creating a bond graph of the cylinder the following denotations are introduced: SE_p - energy effort source which corresponds to pressure p_o, SE_c - effort source which corresponds to Coulomb friction, C_1, C_2 - hydraulic capacitances in cylinder chambers, A - piston area, TF - transformer of hydraulic energy into mechanic energy, I - inertance which corresponds to masses of piston and external loads, R_1, R_2 - hydraulic resistances of throttle valves, R_v - resistance corresponding to viscotic friction between piston and cylinder barrel, proportional to piston velocity v, R_{11}, R_{12} R_{21}, R_{22} - hydraulic resistances dependent on variable flow slots between piston and cylinder sleeve. In bond graph integration component INT are also included. The bond graph of a hydraulic stepper cylinder, for previously determined parameters, is represented in Fig.1c.

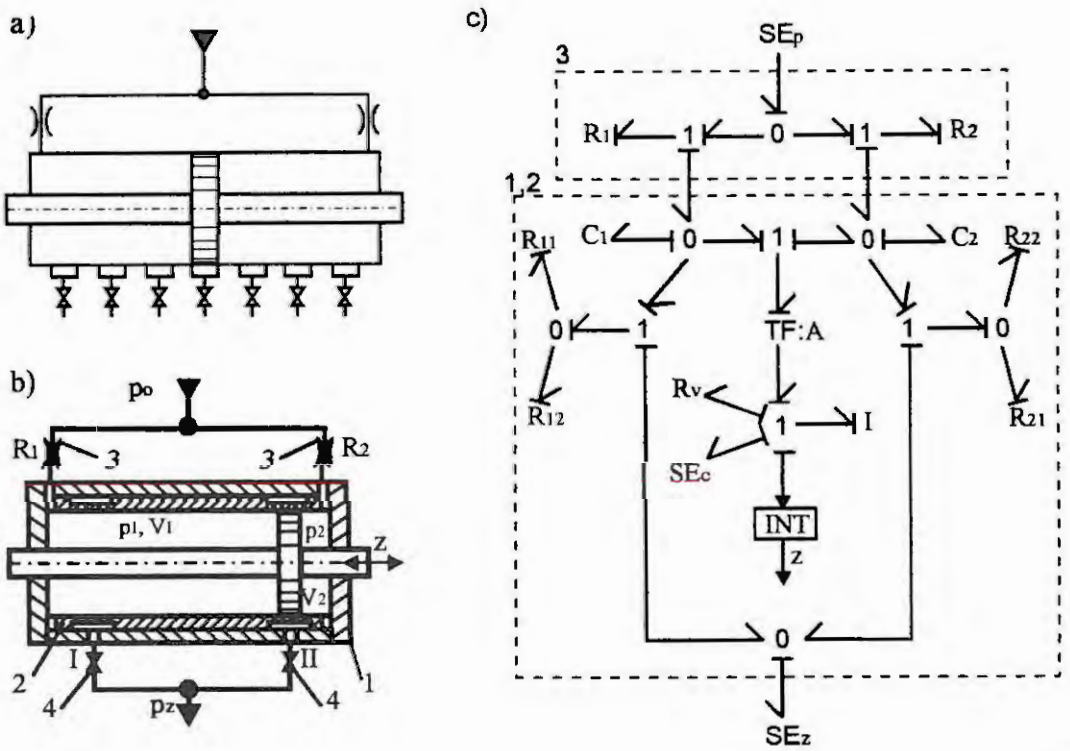


Fig.1. Model of hydraulic stepper cylinder: a) schematic representation, b) experimental model: c) bond graph, 1 - cylinder, 2 - sleeve, 3- throttle valves, 4 - control shutoff valves,

After analysing the created model of the dynamic stepper cylinder it has been decided to extend it by conduits between throttle valves and cylinder. Such extension is justified by the fact that conduits can have significant lengths. The phenomena which occur during damping of piston vibrations and are accompanied by return flow to conduits should be also taken into consideration. In the extended dynamic model the following parameters of conduits will be taken into account: C_{l1} and C_{l2} - hydraulic capacitances, I_{l1} and I_{l2} - hydraulic inertances, R_{l1} and R_{l2} - hydraulic resistances. The bond graph of such an extended model of the dynamic stepper cylinder is represented in Fig.2.

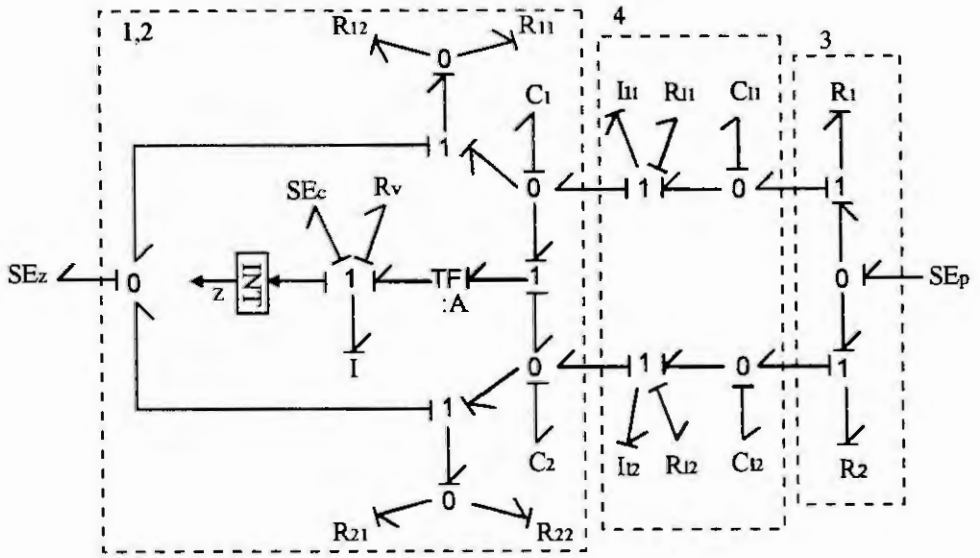


Fig.2. Bond graph of extended dynamic model of stepper cylinder: 1,2 – cylinder and sleeve, 3- throttle valves, 4 - conduits

Bond graph of dynamic model of stepper cylinder with a new element of DB type

In hydrostatic drive system pulsatory flow is connected with self-excited pressure surge, opening and closing of valves, elasticity of elements and operation of generators. In pulsation flow in pipelines constant distribution of parameters takes place. In four-terminal network theory the wave equation is solved by four-pole equations in the matrix form:

$$\begin{bmatrix} p_2(s) \\ Q_2(s) \end{bmatrix} = \mathbf{G}_t \begin{bmatrix} p_1(s) \\ Q_1(s) \end{bmatrix} = \begin{bmatrix} \cosh\left(\sqrt{N(s)}\,Ts\right) & -Z_l\sqrt{N(s)}\,\sinh\left(\sqrt{N(s)}\,Ts\right) \\ -\dfrac{1}{Z_l\sqrt{N(s)}}\sinh\left(\sqrt{N(s)}\,Ts\right) & \cosh\left(\sqrt{N(s)}\,Ts\right) \end{bmatrix} \begin{bmatrix} p_1(s) \\ Q_1(s) \end{bmatrix} \tag{1}$$

where: $\mathbf{G}_t$ – transfer function, T – delay time, Z_l – characteristic impedance, I_l – an interna effect, C_l – hydraulic capacitance, $N(s)$ – frequency dependent friction factor.

Modelling of pulsating flow in pipelines by means of bond graphs causes difficulties as the graphs have been adapted for modelling the dynamics of concentrated parameter systems. In order to include constant distribution of parameters in pipelines a new element of bond graphs called double bond – DB is introduced. To make it distinct from the graphic symbol used for simple bond, double bond DB is designated by double line. Along DB element the changes of pressure p and flow Q depending upon $\mathbf{G}_t$ – functions take place. For this case at the beginning of the element DB is 1– junction with impedance Z_1, and at the end 0 – junction with impedance Z_2. Fig.3 presents a bond graph of stepper cylinder with new elements. DB.

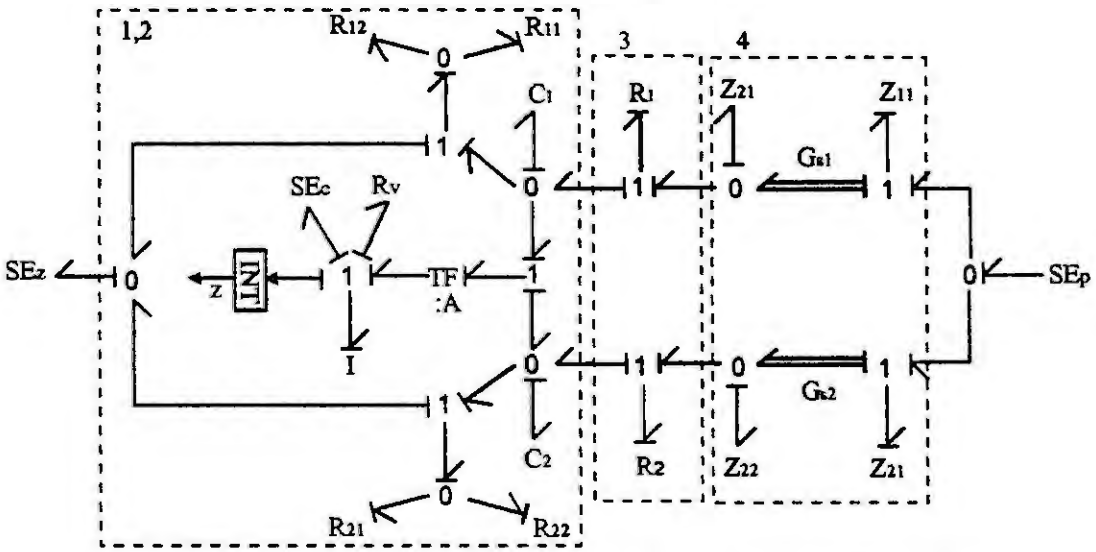


Fig.3. Bond graph of dynamic model of stepper cylinder with a new elements DB:
1,2 – cylinder and sleeve, 3- throttle valves, 4 –supply pipelines

From equation (1), after taking into consideration the direction of wave reflection and resistance R_k at the end of pipelines ($p_2 = R_k Q_2$), the impedance $Z_l(n\omega)$ in pipelines for the nth harmonic wave ($n = 0,1,2,...,\infty$) is obtained for ith pipelines:

$$Z_{1i}(n\omega) = \frac{p_{1i}(n\omega)}{Q_{1i}(n\omega)} = Z_l \frac{\phi\left[1+\mathrm{tg}^2(Tn\omega)\right] + j\left(\phi^2-1\right)\mathrm{tg}(Tn\omega)}{\phi^2+\mathrm{tg}^2(Tn\omega)} \tag{2}$$

where: k – wave propagation coefficient, and ϕ – resistance coefficient of the pipe (for $R_k > Z_l$ $k = p_2/p_1 > 0$ and $\phi = Z_l/R_k < 1$; for $R_k < Z_l$ $k = p_2/p_1 < 0$ and $\phi = Z_l/R_k > 1$).

Digital simulation

The dynamic characteristics of a stepper cylinder can be determined by the method of digital simulation on the basis of the dynamic model represented by means of bond graph in Fig.3, using one of the available simulation programs, e.g. CSSL. In digital simulation the following parameter values were introduced: $A = 0.77\ 10^{-3}$ m^2, $I = 12$ kg, $SE_c = 100$ N, $p_o = 15$ MPa, $R_1 = R_2 = 0.41\ 10^9$ Pas/m^3, $\eta = 0{,}062$ Pas, $E_c = 895$ MPa, $C_1 = C_2 = 0.85\ 10^{14} \div 0.42\ 10^{13}$ m^3/Pa, $\rho = 850$ kg/m , $I_1 = 1.1$ MPas^2/m^5, C_l 0.8 10^{-14} m^3/Pa, R_l 1.28 10^9 Pas/m^3. After reverse control of shutoff valves I and II, when the cylinder piston changes its position dynamic characteristics are determined to show runs of pressure difference $p_1(t) - p_2(t)$ in cylinder chambers and displacements $z(t)$ of cylinder piston. The exemplary dynamic characteristics are presented in Fig.4. Requirements for a stepper cylinder involve a high precision of piston positioning and strong and possibly short-time damping of its vibrations.

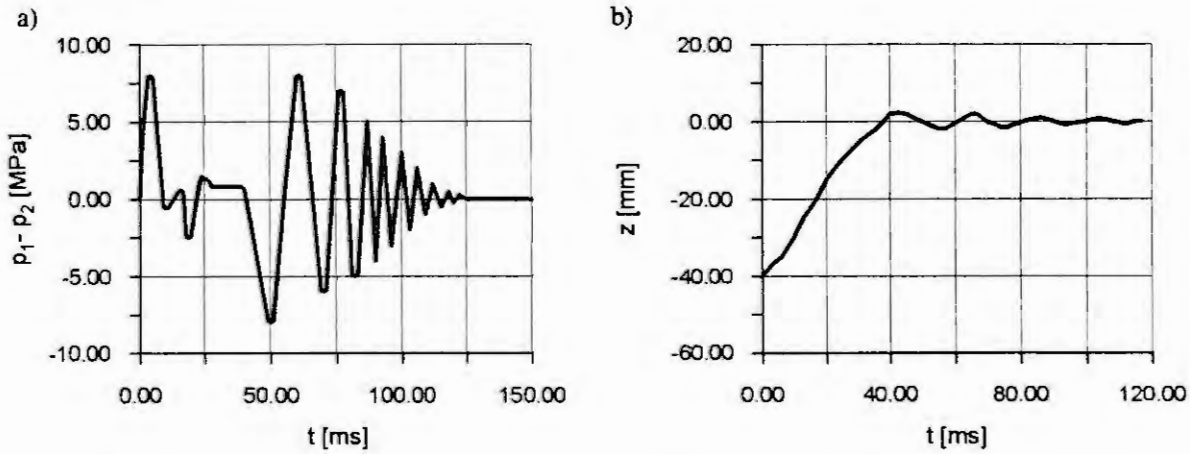


Fig.4. Dynamic characteristics of hydraulic stepper cylinder: a) pressure difference, b) displacement

Summary

The design solution of a hydraulic stepper cylinder presented in this paper allows to obtain many different positions of piston by exchanging the sleeve with properly made outlet slots. Bond graphs, applicable to digital simulation, are used to modelling of the dynamics of a hydraulic stepper cylinder. The dynamic model of a stepper cylinder has been extended by conduits and supply pipelines. The dynamic properties of a hydraulic stepper cylinder are confirmed experimentally [1]. The application of bond graphs with a new element called double bond – DB in modelling of pulsating flow in supply pipelines is described. The method used in modelling of the pulsation flow with DB element enables determining of transmittance, impedance and frequency characteristics of single pipelines. Double bond DB can be linked to other elements of bond graphs.

References

[1] Dindorf, R., Selected problems of modelling of hydraulic systems dynamics (in Polish). Monography 189. Cracow University of Technology, Cracow, 1995.

[2] Dindorf R., Modelling of the dynamic of a hydraulic positional cylinder, Engng. Trans., 44 (1996) 1.

[3] Dindorf R., Possibilities of improving stedy-state characteristics of hydraulic pressure-control valves, Archiv. of Mech. Engng, 35 (1988) 3.

[4] J.M. Thoma, Simulation by Bondgraphs. Springer Verlag, Berlin, New York 1990.

IMPACT OF PHYSICAL ANALOGIES ON CHOICE OF POWER CO-VARIABLES

A.C. Fairlie-Clarke

The University of Glasgow, Glasgow G12 8QQ, Scotland.
Department of Mechanical Engineering. E-mail: tonyfc@mech.gla.ac.uk

Abstract. The paper shows that the choice of power co-variables is not arbitrary, but has physical significance. Force is the flow variable, not velocity. The potential/flow description is presented and its advantages are demonstrated. It allows a single procedure to be used to create bond graphs for systems in all domains, and an understanding of its principles helps to remove the confusion often caused by bond graph semantics. Bond graphs from all domains can be described as comprising common potential junctions that define the flow path and common flow nodes to which all elements of the system are attached.

Introduction

Two primary graphical schemes for modelling dynamic systems in terms of energy flow are linear graphs and bond graphs [1]. The former are drawn using the terms 'across' and 'through' to describe the power co-variables, while bond graphs have commonly adopted the descriptions of 'effort' and 'flow'. The two descriptions are equivalent, with the across variable corresponding to the effort variable, except in the mechanical domain where velocity is the across variable but force, not velocity, is taken as the effort variable. There is mathematical symmetry between the power co-variables, so the assignment of a particular name to a variable does not affect the validity of the solutions. However, the semantics do affect the generalised representation of systems across different domains, and this is significant because an advantage of bond graphs is that they can be used to model multi-domain systems, so analogies are important. The effort/flow pairing is well established as the convention with bond graphs. Some authors [2,3] have recognised advantages for the across/through co-variables, but bow to the convention and take refuge in the mathematical symmetry.

This variance in the definition of the power co-variables in the mechanical domain serves to confuse many who might otherwise adopt bond graphs as an aid to system modelling. It is desirable to resolve the issue and bring bond graphs into the mainstream of approaches for modelling of dynamic systems. However, arguments continue to be put forward for both conventions [4,5]. This paper provides a weight of argument in support of the across/through co-variable pairing, and for naming these the potential and flow variables. The mathematical symmetry between co-variables is not matched by a physical symmetry, and the use of force (or torque) as the effort co-variable and velocity as the flow co-variable involves an incorrect description of the physical behaviour of mechanical systems, which is accommodated in bond graph methods only at the expense of increased complexity.

In the paper, the main features of the potential/flow description are reviewed. The principal arguments that have been made in support for force as effort are answered, and it is shown that the potential/flow description also allows a single common procedure to be used to create bond graphs for systems in all domains. To support the argument, it is shown that incompressible fluid systems can be modelled in terms of the force/velocity co-variables, and that these can be viewed as primary variables, with a gyration being used to derive the pressure/ volume flow co-variables commonly used in fluid systems.

The Potential / Flow Description

Fairlie-Clarke [4] defines the dynamics of various system domains in terms of power co-variables described as the potential variable and the flow variable. For a translational mechanical system these are velocity and force respectively. In an electrical system they are potential difference (voltage) and current, and in an incompressible fluid system they are pressure and volume flow. To achieve consistency in descriptions in the various domains, the flow variable must be associated with the flow of some matter that can divide between parallel parts of the circuit, can have a source and a sink, and can be stored. Velocity quite clearly fails this test and cannot be described as a flow variable, whereas force fully satisfies the test if viewed as the rate of flow of a mechanical charge, where the mechanical charge stored in a mass is equal to its momentum. However, velocity is not readily viewed as an effort. Thoma [3] presents potential as an alternative view of effort, and this term is preferred since, if two masses travelling at different velocities are connected by, say, a spring, then a force will flow between them. Thus a difference in velocity provides a potential for force to flow.

With this description, the basic system elements are flow stores having capacitance properties, potential stores having compliance properties, and energy dissipaters having resistance properties. Fairlie-Clarke [4] distinguishes between the absolute capacitance of mass elements, which absorbs flow, and the difference capacitance of electrical capacitors, which builds up through the passage of flow. The resistive property

diminishes the flow rate resulting from a given potential difference. The intuitive interpretation is that the greater the resistance the more the effort needed to sustain movement. In the mechanical domain this interpretation relates to force as the effort variable. In adopting the force as flow description, a new interpretation is needed that relates to resistance to force. In this case low damping creates high resistance, with a high relative velocity necessary to establish a force across the damper. This is not a comfortable viewpoint, but it results in a good analogy between the electrical, mechanical and fluid domains in that a void between elements equates to infinite resistance. The compliance property is descriptive of the reversible response of an element to a potential difference whereby it enters a stressed, or deformed, state in which the flow across the element depends on the level of stress, and not on the potential difference across the element.

Force as Flow versus Force as Effort

The above arguments should suffice to settle the matter in favour of force as flow, but historically they have not done so. Many bond graph practitioners argue that the mathematical symmetry makes the choice arbitrary, while Hogan and Breedveld [5] present a number of arguments in support of pressure as an effort variable (which implies that velocity is a flow variable). These arguments are addressed by Fairlie-Clarke [6], but only a few issues raise serious points..

1. The principal argument by Hogan and Breedveld is that pressure is intuitively analogous to force, and therefore both should be effort variables. For many people this is true, but it is only a point of view. An equally valid view is that pressure is caused by compression, which is a consequence of velocity. Viewing pressure and velocity as potential variables reinforces this view.
2. The argument for force as effort also cites the fact that the kinetic energy of a solid mass and a fluid mass are analogous. Of course this is true, and it presents a quandary when viewing force as flow since the kinetic energy of a solid mass is then derived from an accumulation of flow, while the kinetic energy of a fluid mass is derived from an accumulation of potential. The answer lies not in the nature of kinetic energy, but in the assumptions made when modelling fluid systems. The behaviour of a fluid system is determined by the conservation of mass and energy, and by Newton's second law, just as for a system of solid elements. Thus incompressible fluid systems can be modelled at a fundamental level as lumped mass elements acted on by forces and moving with a certain velocity. Pressure and volumetric flow can then be derived by a gyration using the inverse of area as the constant of proportionality. The physical behaviour of the system does not, of course, gyrate, so the convenience of using the derived variables comes at the cost of some loss of analogy.
3. A physically valid description of systems should distinguish between elements providing steady state storage of energy as opposed to equilibrium storage. The potential/flow description has a mass as analogous to an electrical capacitor, and both exhibit steady state storage of energy since both will retain their stored energy if disconnected from the circuit. Hogan and Breedveld's argument that an inductance exhibits steady state storage of energy does not stand up under this test.

A further argument in support of the potential/flow description stems from the next section. Use of the effort/flow model means that bond graphs for mechanical systems must be drawn using slightly different procedures to electrical systems [1]. However, if the potential/flow model is adopted, then a single procedure can be used to draw bond graphs for any system, and bond graphs are then amenable to a single rational explanation.

Construction of Bond Graphs

1. Place one harpoon pointing into the system to represent the start of the flow of energy at a boundary point.
2. Trace the flow through the circuit. Place a '0' junction and attach a harpoon to show the flow of energy diverted to each absolute capacitance element. Place a '0' junction and attach a harpoon to show the flow of energy diverted along each tributary of the circuit that terminates at a boundary connection. Place a '0' junction and attach a harpoon to show the flow of energy diverted along each byway, and add another '0' junction when it rejoins the circuit.
3. Place a '1' junction and attach a harpoon to show the flow of energy associated with each element that is not an absolute capacitance.

The bond graph now represents a direct mapping of all the properties of all the elements in the system. This is the full bond graph, but it can be reduced. However, a reduced bond graph may not provide a good visual representation of the system, and it cannot be used to reproduce the schematic.

1. Any harpoons showing the flow of energy to a point of common ground potential can be removed because there will be no energy flow.
2. All elements through which there is a common flow can be attached to a single '1' junction.
3. '0' junctions that serve to join the flow along parallel paths back into the main circuit can be removed, provided that there are no remaining elements attached to the circuit downstream of the junction.

4. Parts of the system that run along parallel flow paths can be treated as a sub-system and attached to a '1' junction on the main flow path, where the flow is equal to the total flow through the parallel paths.

Redundancy in the annotation of bond graphs can be avoided by replacing the '1' and '0' junction symbols with the symbol representing the variable that remains common through the junction. This also enables a more natural visual interpretation of the bond graph since no mental translation has to be made of the meaning of the junctions. With the potential/flow description, bond graphs can be described as comprising common flow nodes, to which are attached all the elements on a common flow path, and common potential junctions from which the flow paths emanate. Flow paths must terminate either at ground (which need not be specifically represented) or at another common potential junction. Any boundary points not at the ground potential must be shown as sources of energy.

Examples

The application of this procedure, and the good representation of analogies given by the potential/flow description, is illustrated by the bond graphs developed for the analogous electrical and mechanical systems shown in Figures 1 and 2. The analogy is not exact because there is a flow to ground through the electrical capacitors, but not through the mechanical mass elements. This difference is represented in the full bond graphs shown in the centre of the figures, but the distinction is lost in the reduced bond graphs on the right, which are exactly analogous for the two systems. The parallel flow paths have been retained in the reduced bond graphs because they provide a better visual representation of the system and an economy of notation. Use of the symbol for the common variable to replace the '0' and '1' junction symbols means that identical bond graphs could be derived using the effort/flow description. However, trying to represent velocity as the flow, when it cannot define a flow path, defies logic and complicates the procedure for drawing the bond graphs.

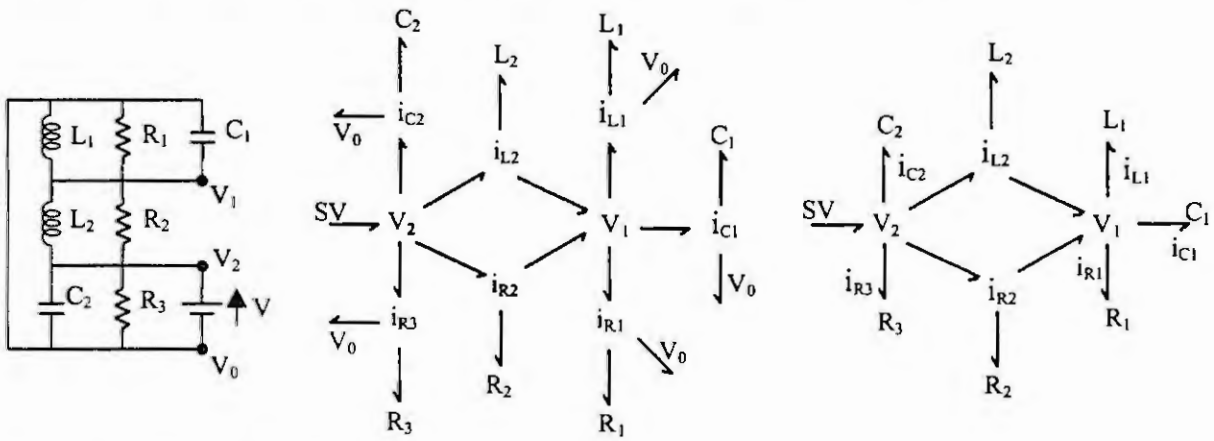


Figure 1. Electrical RCL System

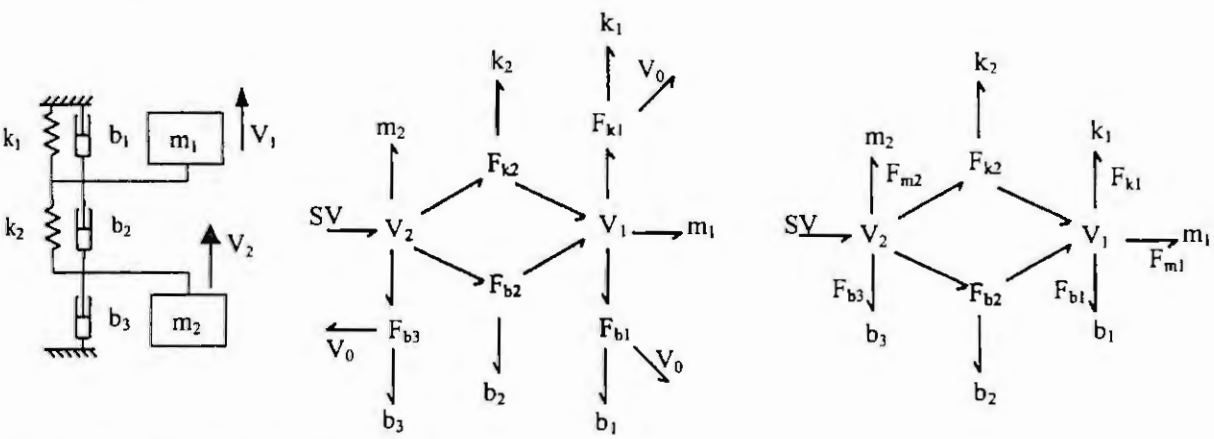


Figure 2. Spring / Mass / Damper System

Figure 3 shows a pictorial view of a series of mass elements being pushed along the ground. The schematic in Figure 4 shows the capacitance and compliance properties of the mass, and the damping due to friction against the ground. The same schematic can also represent lumped masses of fluid flowing along a parallel pipe. Thus the bond graph in Figure 5 represents either a mechanical or a fluid system. In the case of the fluid system, the pressure and volumetric flow can be derived directly from the velocity and force at any point by using a gyration with the inverse of the pipe cross section area as the operator. Figure 6 shows a fluid pipe with varying cross section, while its bond graph is shown in Figure 7. In this case a pair of gyrations are used at the boundary of each lumped mass to give incremental changes representative of the continuous variation in force and velocity caused by the varying pipe section. The increment in force that occurs through the gyrations is the longitudinal component of the pressure reaction force exerted by the pipe wall.

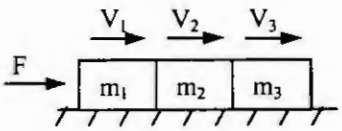


Figure 3. Lumped Mass System

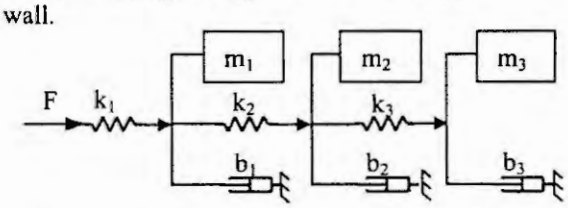


Figure 4. Properties of the Lumped Mass System

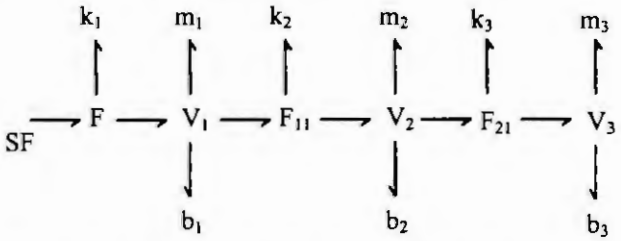


Figure 5. Bond Graph of Lumped Mass System

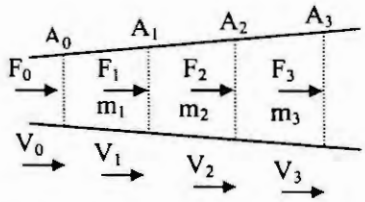


Figure 6. Fluid Flow in Pipe

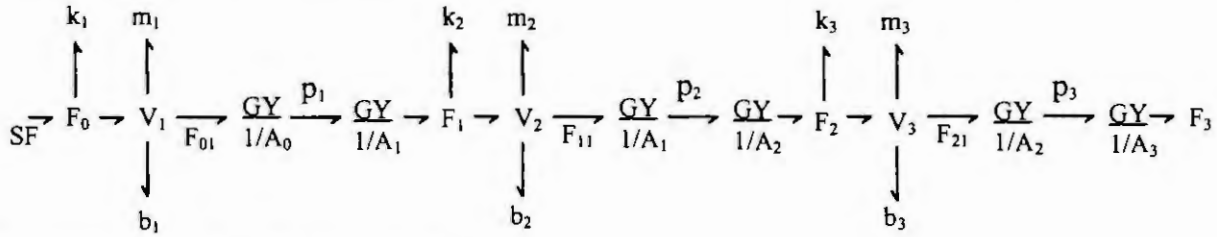


Figure 7. Bond Graph of Fluid Flow in Pipe

Conclusions

The choice of the power co-variables is not arbitrary, but has physical significance. Velocity is not a flow variable, and representing it as such leads to complications and confusion. The force as flow description is logical. It allows easier preparation of bond graphs and clarifies the analogies between system domains once the concept of this flow, and of resistance to it, become familiar. It helps if it is recognised that gyrations and transformations between power co-variables in different domains are usually descriptive rather than causal.

References

1. Karnopp, D.C., Margolis, D.L. and Rosenberg, R.C., System Dynamics: a Unified Approach, 2nd edition, John Wiley, New York,1990.
2. Cellier, F.E., Continuous System Modelling, Springer-Verlag, 1991.
3. Thoma, J.U., Introduction to Bond Graphs and their Applications, Pergamon Press Ltd.,1975.
4. Fairlie-Clarke, A.C., Force as a flow variable. Proceedings of the Institution of Mechanical Engineers, Volume 213, Part I, 1999.
5. Hogan, N. and Breedveld, P.C., The physical basis of analogies in network models of physical system dynamics. In: Proceedings of the 1999 International Conference on Bond Graph Modelling and Simulation. Simulation Series, Volume 29 (Eds.: J.J.Granda and F.E. Cellier), San Francisco, January 1999.
6. Fairlie-Clarke, A.C.. A reconciliation of bond graphs with other graphical models. Proceedings of the Institution of Mechanical Engineers, under review.

SOLVABILITY AND R- CONTROLLABILITY FOR GENERALIZED SYSTEMS MODELLED BY BOND GRAPH

A. MOUHRI[1], A. RAHMANI & G. DAUPHIN- TANGUY
Ecole Centrale de Lille, LAIL, UPRESA CNRS 8021,
BP 48. F59651. Villeneuve d'Ascq Cedex FRANCE.
[1] Phone no. (33) 320335415 – email. mouhri@ec-lille.fr

Abstract. In this paper, a bond graph interpretation of the generalized characteristic polynomial of (*sE-A*) which plays a key role in solvability of the generalized system, is provided. This graphical method is based exclusively on causality handling and causal cycle families gains. Next, a method applied directly on bond graph model who gives structural rank of generalized state matrix is developed. Based on this previous results, a procedure for the formal determination of the infinite modes number who singular system can exhibit, is proposed. Otherwise, starting from known algebraic criteria, the structural r-controllability is derived only on bond graph model.

Introduction

Let us consider the general linear time invariant multivariable system, described by the state equation (1)

$$\begin{cases} E\dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) \end{cases} \qquad (1)$$

with the state vector $x \in \Re^n$, the input vector $u \in \Re^m$ and the output vector $y \in \Re^p$.

Generalized systems of the form (1) with E singular matrix, currently appear as models of large scale systems, circuits, economics... In literature, we find lot of synonymous terms used to define this type of state space model: singular system, generalized state space system, implicit system, differential/algebraic system or descriptor system. The analysis of their structural properties has received great attention. Several approachs works were done to study structural controllability. The first one is the graph approach: Reinschke [7] gave from known algebraic criteria, graphical conditions for the different types of structural controllability. The second one, the geometric approach, has been extended to descriptor systems by Wonham [9].

In this paper, a procedure which allows an easy determination of impulsional modes of a descriptor system is presented. Structural analysis by bond graph means of generalized system rank and of R- controllability property is studied. Then, some conclusions are derived directly from bond graph model topology without actually deriving state equations but by using the causal coupling concept.

This paper is organized as follows: first section is devoted to the recall of preliminary results concerning bond graph models with dynamic elements in derivative causality. The main result is given in second section, which is gathered with the formal calculus of generalized characteristic polynomial of (*sE-A*) based on causal cycle families gains. The theorem gives the rank of generalized state space matrix and analysis of structural controllability is developed in section 3. Finally, an example illustrates the application of the results.

I – Generalized state space or implicit model : bond graph approach

A unified representation such as the bond graph model leads to the calculation of the state equation whatever the physical domain. Algebraic loops between R-elements or derivative causalities in a bond graph model involve some problems which may induce manipulations on the equations to avoid simulation difficulties.

The junction structure of a bond graph model contains information on the types of elements which compose it, and on the manner they are interconnected. The different vectors involved in a bond graph are linked through the junction structure equation as:

$$\begin{pmatrix} \dot{x}_i \\ z_d \\ D_i \\ Y \end{pmatrix} = \begin{bmatrix} S_{11} & S_{12} & S_{13} & S_{14} \\ S_{21} & S_{22} & S_{23} & S_{24} \\ S_{31} & S_{32} & S_{33} & S_{34} \\ S_{41} & S_{42} & S_{43} & S_{44} \end{bmatrix} \begin{pmatrix} z_i \\ \dot{x}_d \\ D_o \\ u \end{pmatrix}$$

Elementary laws are associated with components:

$$z_i = F_i x_i, \quad x_d = (F_d)^{-1} z_d \text{ and } \quad D_o = L D_i$$

x_i and x_d are the state vectors associated with the I and C elements respectively in integral and derivative causality. z_i ,z_d are the complementary state vectors, and F_i (resp. F_d) is a diagonal matrix composed of

parameters associated with I and C elements. These matrices are always invertible for bond graph models without multiport elements.

D_i and D_o denote the vectors composed of variables flowing respectively into and out of the R-components ; L is a diagonal matrix composed of resistance and conductance parameters.

- In a minimal representation, where the state vector is composed only of (I, C) elements in integral causality $(x = x_i)$, the matrix E is non singular. The system is said to be regular.
- In a complete representation, the state vector can be chosen as: $x=\left(x_i^t \quad x_d^t\right)$, $x_i \in R^{n_i}$, $x_d \in R^{n_d}$, and with $n = n_i + n_d$, n_i (resp. n_d) the numbers of (I, C) elements in integral (resp. derivative) causality when an integral causality assignment is performed on the bond graph model.

A particular representation with E not full rank is imposed by the bond graph model. Equation (1) can be decomposed as follows:

$$\begin{pmatrix} \mathbf{I}_{n_i} & \mathbf{E}_{id} \\ 0_{n_d x n_i} & 0_{n_d x n_d} \end{pmatrix} \begin{pmatrix} \dot{\mathbf{x}}_i \\ \dot{\mathbf{x}}_d \end{pmatrix} = \begin{pmatrix} \mathbf{A}_i & 0_{n_i x n_d} \\ \mathbf{A}_{di} & \mathbf{A}_{dd} \end{pmatrix} \begin{pmatrix} \mathbf{x}_i \\ \mathbf{x}_d \end{pmatrix} + \begin{pmatrix} \mathbf{B}_i \\ \mathbf{B}_d \end{pmatrix} \mathbf{u} \qquad (2)$$

Some hypothesis are taken in consideration :

- two dependent (I, C) elements in derivative causality are not directly causally connected (it is possible to simultaneously change their causality to obtain two dynamical elements in integral causality), $S_{22} = 0$.

- it is impossible to have a derivative causality on elements I or C causally connected to linear R- elements (it is possible to exchange the causalities), $S_{23} = 0$.

The expression of submatrices in proposed form (2), are deduced from the junction structure matrix built from the bond graph model and elementary physical laws. For 1-port bond graph model, the formal procedure to obtain different matrices in the form (2) directly on bond graph are already developed explicitly in [8]. For sake of simplicity, we recall the following expressions :

$$\begin{array}{lll} E_{id} = -S_{12} & A_{dd} = -F_d & A_i = [S_{11} + S_{13}L(I - S_{13}L)^{-1}S_{31}]F_i \\ A_{di} = S_{21}F_i & B_d = S_{24} & B_i = S_{14} + S_{13}L(I - S_{13}L)^{-1}S_{34} \end{array} \qquad (3)$$

II - Structural Solvability of generalized systems arising from bond graph models

The solvability is one of the main structural properties which every study must begin with, their important role appears precisely when studying infinite structure from the transfer function. It has been discussed for ($B = I$) in Gantmacher [4] and Wilkinson [5], and Luenberger [6] generalises their results to arbitrary B.

1. Explaining

The solvability of model (1) is defined as the existence of a unique solution for any given control function sufficiently differentiable $u(t)$ and any given admissible initial condition corresponding to the given $u(t)$. The use of Laplace transform leads to write from (1) : $(sE-A)\,X(s) = EX(0) + BU(s)$. A necessary and sufficient condition of the existence and unicity of the solution is that the characteristic polynomial det $(sE - A)$ is different from zero for almost every $s \in \mathbb{C}$.

2. Formal calculus of det (sE-A)

The characteristic polynomial is equal to the denominator of the transfer function $T(s) = C(sE - A)^{-1}B$, expressed as: $P_\Sigma(s) = |sE - A| = \sum_{i=0}^{n} p_i s^{n-i}$. The formal calculus of parameters $p_i, i = 0,...,n$ is based exclusively on causality handling and causal cycle families concept. Hence, it requires a new definition of causal cycle family order and gain.

Definition *the causal cycle family is a set of disjoint causal cycles. This family is said to be of order* t*, if it contains* n_i *independent dynamical elements and* n_d *statically dependent elements with* $t = n_i - n_d$*. The causal cycle family gain is equal to the product of the different causal cycle gains which composed this family. It is denoted* G^t *when t is their order.*

Theorem *every coefficient p_i of $\mathbf{P_\Sigma(s)}$ is given by the following expressions:*

* for each $i = 0,1,\ldots,n$ and $i \neq n_d$ $$p_i = \frac{\sum_j (-1)^{d_j} * \tilde{G}_j^{\,i-n_d}}{\prod \tilde{g}(x_d)} \qquad (4)$$	* If $i = n_d$, the expression of coefficient p_{n_d} is related to causal cycle families of order zero : $$p_{n_d} = \frac{1 + \sum_j (-1)^{d_j} * \tilde{G}_j^{0}}{\prod \tilde{g}(x_d)} \qquad (5)$$

where "j" is the jth causal cycle family of order $\mathbf{i - n_d}$, d_j *is the number of disjoint causal cycles constituting the jth family and* $\tilde{G}_j^{i-n_d}$ *is the gain constant term of jth causal cycle family of* $\mathbf{i - n_d}$ *order.*

$\tilde{g}(x_d)$ *is the constant term in transmittance of elements {I, C} affected with derivative causality in the bond graph model:* $\tilde{g}(I_d) = I_d$ *and* $\tilde{g}(C_d) = C_d$.

III - Structural properties of generalized systems arising from bond graph models

The structural analysis of the properties of linear generalized systems such as structural controllability found an interest for automation of control design. These properties can be pointed out before calculating mathematical representations. By using these concepts, we may reflect the difference between singular systems and regular ones (classical systems).

1. Rank of generalized state space

In order to show how a bond graph model of a descriptor system is a good tool for the symbolic calculus and structural analysis of the associated mathematical model, we begin by studying the generalized rank of the state matrix A which allows the determination of the minimum number of inputs sources necessary for control and the optimisation of their positions to simplify state feedback control laws.

Property 1 *the bond graph rank of the generalized state matrix A denoted by* ***bg-rank (A)*** *is equal to* $(n_i - q) + n_d$, *with* n_i *(resp.* n_d*): the number of dynamical I,C- elements in integral causality (resp. in derivative causality, q : the number of I,C elements staying in integral causality when a derivative causality is applied on the bond graph model.*

Proof. The matrix theory (formula Schur [4]) gives the following equality : $\det(A) = \det(A_{dd}).\det(A_i)$, the A_{dd} matrix is invertible because it is equal to $-F_d$, diagonal matrix composed of the parameters associated with *I* and *C* elements in derivative causality for a system modelled by bond graph without multiport elements. Hence degeneracy of *A* i.e. its rank less than *n*, is due to degeneracy of the matrix A_i which is classical state matrix associated with bond graph without dependent elements.

Property 2 *the structural rank of the singular matrix E (noted r) is equal to the number of dynamical elements (I,C) in integral causality when an integral causality assignment is performed on bond graph model: rank(E)=* n_i

2. Structural Controllability

For generalized system, the controllability property is decomposed in :

- R- controllability : related to the capacity to control the finite dynamical modes (traditional controllability of exponential modes for regular system). It is associated to the differential part composing the state space.
- Impulse controllability which shows the ability to remove the infinite modes.

The reachable set is defined as a subset of $\Re^n$ comprising all consistent initial values $x(0)$ [2]. The different kinds of controllability have been defined as follows (cf. [2, 7]):

Definitions a generalized system is said to be:

* R - controllable if it's controllable within the reachable set,
* Impulse - controllable if all impulsive modes can be excited by suitably chosen no-impulsive inputs.
* Completely - controllable if it is controllable within $\Re^n$.

Necessary and sufficient conditions for controllability have been proved (see [10]):

Lemma a generalized system (1) is said to be:

* R - controllable iff rank $(sE - A, B) = n$ for all $s \in \mathbf{C}$
* impulse - controllable iff $\operatorname{rank}\begin{pmatrix} E & 0 & 0 \\ A & E & B \end{pmatrix} = n + \operatorname{rank} E$

* completely - controllable iff rank $(E, B) = n$ and rank $(sE - A, B) = n$ for all s.

The objective is to present a method using bond graph methodology to derive information on structural controllability. Currently, we restrain the study to R-controllability which is expressed in bond graph terms by the following theorem:

Theorem A generalized model is structurally state R-controllable iff the two conditions are verified :

(i) all (I,C) elements are connected with a source.

(ii) no dynamical element remain in integral causality when :
- a derivative causality assignment is performed, and
- a dualisation of the maximal number of input sources is performed in order to suppress these integral causalities.

Proof. this theorem is an extension of the one associated with regular model.

Before studying the impulse controllability property, it is necessary to show if a model presents or not impulsive modes. For that, we developed a procedure which derives from previous results :

3. Infinite modes :

In the case where the pencil $(sE-A)$ is regular, the output and the input under null initial conditions $(Ex(0) = 0)$ are related by the non strictly proper transfer function, as follows: $Y(s) = [C(sE-A)B\,]U(s)$. But in generalized case, the choice of the initial vector $x(0)$ must be not arbitrarily and must verify some conditions of consistency. The response of the system can exhibit d exponential modes where d is equal to degree of $det(sE-A)$. Let r = rank(E), the actual order of system hence $d \leq r$, the equality occurs in the regular case (i.e. E is not singular). The free- response of the descriptor system, i.e., $x(t)$ for $t \geq 0$ when $u(t) = 0$, consists of combinations of d exponential modes, characteristic frequencies for which $(sE-A)$ is singular (d = degree of characteristic polynomial). In addition, however, it contains $(r- d)$ infinite-frequency modes : impulsive modes corresponding essentially to $(sE-A)$ losing rank at $s = \infty$ hence to poles at infinity (see [1], [2],[3]).

Calculus procedure of infinite modes number

This algorithm allows us to formally determine the number of impulsive modes the system response highlights, without any complex calculus :

Step 1: Determine the rank (noted r) of the singular matrix E .

Step 2: Determine d, degree of the characteristic polynomial (finite modes number) linked to the first non null coefficient p_i , $i \in \{0 \cdots n\}$; it corresponds to consider the causal cycle families of order $i - n_d$. Hence, this degree is $d = n - i$.

Step 3: if $d < n_i$, there has $n_i - d$ infinite modes.

Theorem In 1-port bond graph models, it is impossible to have impulsive modes.

Proof. Let us apply the decomposition method by the Smith form which clearly reflects the physical meaning of singular system and gives two independent parts. The first equation is a differential one and the second is an algebraic equation that represents the connection between subsystems.

From matrix theory, there exist non-singular matrices $P = diag(I_{n_i}, I_{n_d})$ and $Q = \begin{pmatrix} I_{n_i} & S_{12} \\ 0 & I_{n_d} \end{pmatrix}$ such that $PEQ = diag(I_{n_i}, 0_{n_d \times n_d})$. By taking the co-ordinate transformation $Q^{-1}x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, $x_1 \in R^{n_i}, x_2 \in R^{n_d}$ the system is restricted system equivalent to followed form :

$$\begin{pmatrix} I_{n_i} & 0_{n_i \times n_d} \\ 0_{n_d \times n_i} & 0_{n_d \times n_d} \end{pmatrix} \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} A_i & A_i S_{12} \\ -S_{12}{}^t F_i & -(S_{12}{}^t F_i S_{12} + F_d) \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} B_i \\ S_{24} \end{pmatrix} u$$

In this case, since $(S_{12}{}^t F_i S_{12} + F_d)$ is also a non singular matrix because (F_d , F_i) are strictly defined positive matrices.

IV - Application

The bond graph model presented figure2 contains one dependent storage element.

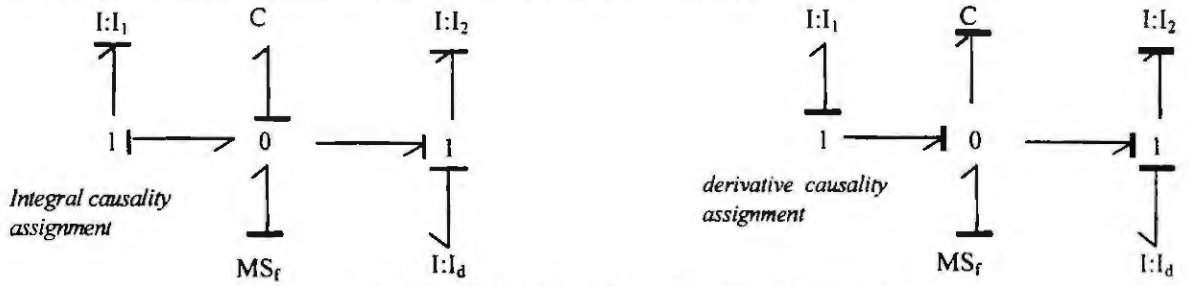


Fig. 2 : *BG with one dependent storage element*

- Generalized state space : all terms of matrices A, B and E are obtained by formal expressions (see [9]) developed for bond graph model with dependent elements :

$$A = \begin{bmatrix} 0 & 0 & -1/C & 0 \\ 0 & 0 & 1/C & 0 \\ 1/I_1 & -1/I_2 & 0 & 0 \\ 0 & 1/I_2 & 0 & -1/I_d \end{bmatrix} \quad E = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

- Solvability : the characteristic polynomial coefficients are calculated by applying previous theorem

$$|sE - A| = \frac{1}{I_d}\left[s^3\left(1 + \frac{I_d}{I_2}\right) + s^1\left(\frac{1}{I_1 C} + \frac{1}{I_2 C} + \frac{I_d}{I_1 I_2 C}\right)\right]$$

- R- controllability :

→ Rank of A : when assigning a derivative causality, all the (I, C) elements in integral causality change their causal stroke position expect I_2 which keeps its integral causality. Element I_d is a dependent storage element, bg-rank (A) = (3-1) + 1 = 3. Then, one control source in suitably position is sufficient to control system.

→ The system is not R- controllable by input source MS_f because the element I_2 stays in its initial causality (integral) when applying a derivative causality on bond graph model and when the flow source is dualized.

Remark If we place the control input (effort source) at the junction 1, the system is R-controllable.

- Impulsive modes : $p_1 = 1 + I_d / I_2$, the first parameter of polynomial characteristic is structurally non null, then no infinite mode is pointed out.

V- Conclusion

This paper gives rules to extend the methodology of symbolic calculus and structural analysis by bond graph, developed for regular system to descriptor system. Symbolic calculus of characteristic polynomial and rank procedure for singular system modelled by bond graph are given. Then, a structural analysis of R- controllability property by bond graph tools is studied.

References

1. Compbell, S.C., Singular Systems of Differential Equations. Pitman, London, 1980.
2. Yip, E. & Sincovec, R., Solvability, Controllability and Observability of Continuous Descriptor Systems. IEEE, Trans. Auto. Control, *V*. AC-2, *N*°3, June 1981.
3. Cobb D., Descriptor variable and generalized singularly perturbed systems. Int., Jour. Cont., *V*33, *N*°6, (1981).
4. Gantmacher, F. R., The theory of matrices. 2 (1974), New York: Chelsea.
5. Wilkinson, J.H., Linear differential equations and Kronecker's canonical form. In: Recent Advances in Numerical Analysis, (Eds.: de Boor, C. and Golub, G.H.) New York : Academic, 1978.
6. Luenberger, D.G., Time-invariant descriptor systems. Automatica, 14 (1978), 473-480.
7. Reinschke, K.J. & Wiedemann, G., Digraph characterization of structural controllability for linear descriptor systems. Linear Algebra and its Applications, 266 (1997),199-217.
8. Mouhri, A., Rahmani, A. and Dauphin- Tanguy, G., Symbolic Determination of Generalized State Equation for Singular System Modelled by Bond Graph. In: Proc. CSCC'99, Athens- Greece, 1999, 229 - 234 .
9. Wonham, W.M., Linear Multivariable Control: a geometric approach. Springer- Verlag, Berlin, 1974.
10. Dai L., Singular control systems, Springer – Verlag, New York, 1989.

Modelling Gravitational Wave Detector Suspensions using Bond Graphs

David Palmer[1], Donald J. Ballance[1], Peter J. Gawthrop[1], Kenneth Strain[2] and Norna A. Robertson[2]

[1]Department of Mechanical Engineering, [2]Department of Physics and Astronomy
University of Glasgow, Glasgow, Scotland. G12 8QQ.
Tel: +44 (0)141 339 8855 Email: {dpalmer, D.Ballance, P.Gawthrop}@mech.gla.ac.uk

Abstract: Michelson interferometer based gravitational wave detectors are currently being constructed by groups throughout the world. Ground isolation is achieved by using a series of pendulums suspended beneath each other. This paper discusses modelling of these pendulums using bond graphs, emphasises the advantage of using bond graphs as a core representation from which specific representations can be automatically generated, and considers the appropriate number of lumped elements needed to approximate a continuous system.

Introduction to gravitational waves and detectors

The existence of gravitational waves were predicted by Einstein's General Theory of Relativity (1916). The effect of a gravitational wave is to produce a strain, h, in space. For two masses separated by a length L a suitably polarised wave will produce a strain given by

$$h = \frac{2\Delta L}{L} \tag{1}$$

where ΔL is the change in length. The waves are produced by asymmetric acceleration of mass. Unfortunately, due to the weak nature of the gravitational force, only astronomical events (such as supernovae) produce strains that an Earth based detector could measure. We expect these strains to be in the region of 10^{-17} to 10^{-18}. Hence the need to build highly sensitive instruments. Laser interferometry is ideally suited to the measurement of small displacements and as such a number of detectors based upon the Michelson interferometer are being developed. A Michelson Interferometer is formed by a beamsplitter and two mirrors. Coherent light split into two beams by the beam splitter is incident on the mirrors. Recombination of the light, at the beamsplitter, results in an interference pattern, the form of which depends on the phase difference between the two beams. This phase difference is introduced through the interferometer having unequal arm lengths. When in operation the detector is configured to have a null output. That is, it is maintained on a dark fringe. A gravitational wave passing through the detector causes the relative arm length to change, thus phase modulating the laser light. The effect of this modulation is to produce sidebands about the laser light frequency (carrier frequency). The gravitational wave information is contained within these sidebands. It is this signal, by feedback to one or more of the test masses, that maintains the detector on a dark fringe. This feedback loop is known as the *Global control.* Since we wish to detect extremely small strains, the components that make up the interferometer must be isolated from all potential noise sources. These include seismic, acoustic, thermal and laser noise. Enclosing the interferometer in a vacuum isolates it from acoustic noise. Filtering of seismic noise is primarily achieved by hanging the optics as a pendulum. Above its natural frequency each stage of pendulum affords a (f_0^2/f^2) attenuation from seismic noise. Where f_0 is the pendulum's natural frequency. The effect of using mechanical systems to attenuate high frequency motion results in an amplification of the seismic motion at the mechanical resonant frequencies. Hence these modes need to have low Q factors (high energy loss). However to minimise the effects of thermal noise, within the detection band of the interferometer, materials of very low energy loss (high Q factors) are used, which, if undamped, lead to unacceptably large motions of the suspended optics. As a result there is a requirement for active control. The implementation of feedback to individual pendulums is known as *local control.* To facilitate the design of multiple pendulums, explore the effect of parameter changes and to develop robust control laws requires accurate system modelling.

The modelling philosophy

For any physical system there are many different models that can represent it. Naturally no model can exactly replicate the physical system. The type and complexity of a model will depend upon its end use. The key factor in the development of a model is the level of complexity needed to answer the modeller's

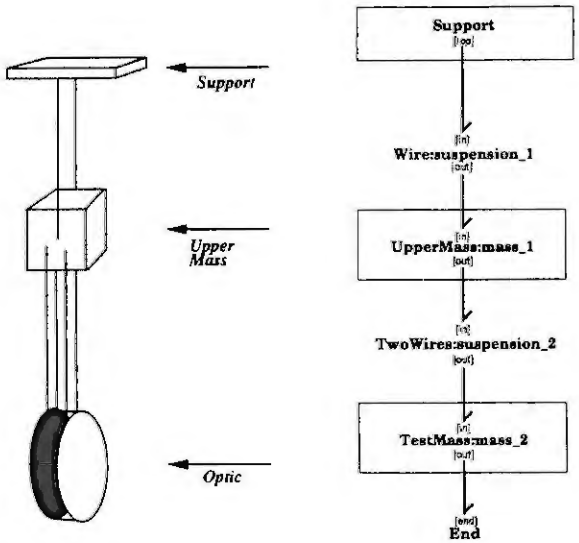


Figure 1: Schematic and Top Level Bond Graph of a Double Pendulum

questions. Too detailed and it may be impossible to analyse and extract essential information from they array of parameters. Too simplified and the model will not reveal essential system information. The aim of the modeller should be to produce the simplest model capable of suppling the relevant information. For example, in the gravitational wave detector, the violin modes are irrelevant in the design of the local control feedback. To include them would produce a higher ordered model/controller than is necessary, making them more sensitive to parameter change. Yet, not to include them whilst developing the global control would result in an unstable controller. Ideally then, a *core* model of a system from which various representations can easily be extracted would facilitate the development of system models of various complexities and for different purposes. Obviously if the core model is hierarchical it enables subsystem components to be changed without having to re-model the whole system. Furthermore, if the modelling technique is unambiguous it can be understood by a computer programme and hence the power of modern computers can be utilised to extract, via model transformations, specific model representations. Hence the use of bond graphs in modelling gravitational wave detector suspensions.

Bond graphs

Bond graphs are an energy based methodology. By using a small set of idealised elements, system models in domains such as electrical, mechanical, and hydraulic can easily be produced. Moreover, since these same elements are used across all domains, multiple domained systems can be readily constructed. Further, it is possible to use the Bond Graph as a core representation of the system from which other representations, such as state space equations and transfer functions, can be generated. Also since the bond graph representation is unambiguous a computer can be utilised to carry out these transformations. The use of bond graphs enables the modeller to construct complex systems in a hierarchical manner. Thus the reticulation of a system into its component parts allows the system model to be constructed from simple subsystems where the physical laws are understood.

Double pendulum modelling

Figure 1 illustrates the schematic and bond graph model of a double pendulum. The bond graph is the top level representation and is itself comprised of some eleven different sub components each of which can be changed independently. From this one model, state space matrices, ordinary differential equations, transfer functions and simulation code etc can all be extracted in formats such as reduce (symbolic algebra), MATLAB and C code. The software package MTT[1] is able to causally complete an acausal bond graph and produce any of these representations.

The benefits of hierarchical modelling will now be demonstrated by examining the sub component Wire. This wire component is itself hierarchically constructed. This facilitates the inclusion, as necessary, of bending and/or violin mode dynamics. Referring to the bond graph of Wire (Figure 2a) the section enclosed by the dashed box determines the angle and linear extension (at that angle) of the wire. Any associated restoring forces are determined by the subcomponent **Beam:wire**. Depending on the nature of this subcomponent either restoring forces, solely due to linear wire extension or, both linear wire extension and wire bending, may be included. Figure 2b contains the subcomponent FourUndamped:violin which models the violin dynamics.

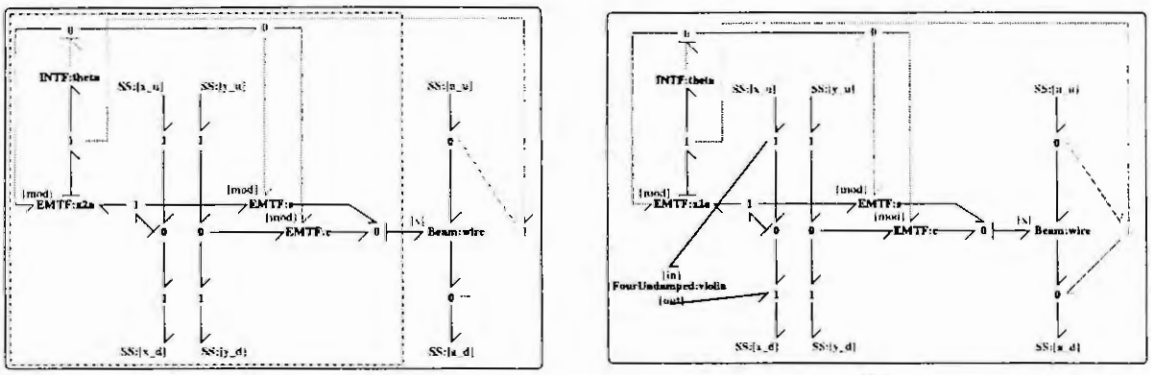


Figure 2: Bond Graph of a Wire (a) without (b) with Violin Modes

Margolis [2] demonstrated that Longitudinal vibrations of a bar maybe represented by the bond graph shown in Figure 3. In the bond graph the number of lumped elements sets the parameters of I and C; $C = \Delta x/EA$ and $I = \rho A \Delta x$), where Δx is the length of the beam divided by the number of lumps, E is Young's modulus, ρ is the mass density and A is the cross sectional area. The accuracy of this method depends on the number of included elements. In the limit, as the number of elements tends to infinity, i.e $\Delta x \longrightarrow 0$ the bond graph model tends to the continuous model 2.

$$E\frac{\partial^2 \zeta}{\partial x^2} = \rho \frac{\partial^2 \zeta}{\partial t^2} \tag{2}$$

Equation 2 is the 1-dimensional wave equation. Transverse waves on a string can, with E and ρ replaced by T and m respectively, be modelled by this equation and hence a bond graph of the same form as Figure 3 can be used to model these modes. T is the wire tension and μ is the mass per unit length.

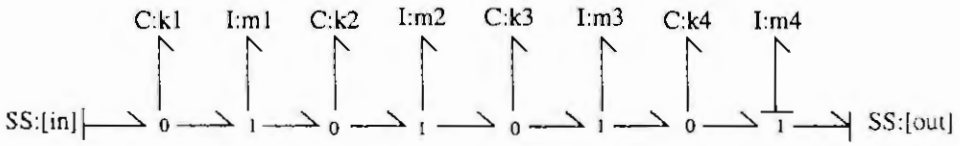


Figure 3: Lumped Bond Graph Replicating Violin Dynamics

$$T\frac{\partial^2 \zeta}{\partial x^2} = \mu \frac{\partial^2 \zeta}{\partial t^2} \tag{3}$$

The question arises as to how many elements should a model include. By increasing the number of model elements the accuracy of the lower frequency increases at a cost of an increase in the number of inaccurate higher frequency modes. However increased accuracy in mode frequencies results in models with a large number of states; a sixteen element wire has 32 states. In Table 1 the number of model elements and the resulting mode frequencies are compared with experimental results and the solution to (3) with fixed end boundary conditions (4)[3].

4 elements	16 elements	simple equ	expt
196.93	201.76	202.08	193.1/205.0/207.6/219.9
363.88	401.58	404.17	not measured
475.43	597.53	606.25	not measured

Table 1: Comparison of mode frequencies produced by a lumped wire bond graph model

$$\nu_n = \frac{n}{2L}\sqrt{\frac{T}{\mu}} \tag{4}$$

The key points here are:

- The bond graph mode frequencies were determined from the A matrix of the state matrices ($\dot{x} = Ax + Bu, y = Cx + Du$). These state matrices were generated from the non linear, causally complete, bond graph model. The model is non linear due to the presence of component constituent relationships containing sines and cosines. Lineraisation was taken about the steady state, i.e state derivatives equal to zero. All transformations were achieved using the software package MTT [1]. This is just one possible use of the bond graph model others include Model-Based Observer Controller design [4] and the generation of C code for model simulation.
- The experimental results were taken from the "four" wires between the upper and test masses. In fact the test mass sits in the bottom of two loops of wire suspended from the upper mass. This accounts for the spread of frequencies.
- Changing the model from 4 to 8 and then 16 elements demonstrates that as the number of elements increases, the bond graph model tends to the continuous model.
- As for how many elements should be included, obviously it is necessary to demonstrate that the bond graph model tends to the continuous model as the elements (theoretically) tend to infinity. However, as is demonstrated by the experimental results, there is some variation in the actual frequency of an individual wire. Therefore, for the purpose of control (as currently envisaged) it is only necessary to include enough elements to ensure that the model's mode frequency lies within the range of the real system mode frequencies.

Conclusions

In this paper it has, through the modelling of a double pendulum and specifically the sub componant Wire, been demonstrated that the bond graph methodology is the ideal tool for system modelling. By using a small number of elements, complex systems, across all energy domains, can be systematically constructed. The hierarchical nature of bond graphs enables small changes to be made without the need for a total re-model. Further, since it is unambiguous, the full power of modern computers can be utilised in the transformation of bond graphs into more recognisable formats. Future work on violin dynamics modelling includes an investigation in to the suitability of the finite-mode model.

References

[1] Peter J. Gawthrop. MTT: Model transformation tools. Online WWW Home Page, 1999. URL: http://www.eng.gla.ac.uk/~peterg/software/MTT/.

[2] D. C. Karnopp, D. L. Margolis, and R. C. Rosenberg. *System Dynamics: A Unified Approach.* John Wiley, 1990.

[3] A. P. French. *Vibrations and Waves.* Chapman and Hall, 1971.

[4] P. J. Gawthrop. Physical model-based control: A bond graph approach. *Journal of the Franklin Institute*, 332B(3), 285–305 1995.

A NEW BOND GRAPH APPROACH TO SENSITIVITY ANALYSIS

Peter H. Roe & Jean U. Thoma
Department of Systems Design Engineering
University of Waterloo
Waterloo, Ontario N2L 3G1, Canada

Abstract. This paper introduces the Sensitivity Bond Graph, which is a special pseudo Bond Graph associated with a system. The efforts and flows in the sensitivity bond graph represent the changes in the efforts and flows in the original, or Parent Bond Graph, of a system, whose R, C and I parameters are themselves subject to variation. The sensitivity bond graph is pseudo because it contains not efforts and flows, e.g. voltages and currents, but the partial derivatives of these variables with respect to some parameter, e.g. T for temperature. Thus the product of 'effort' and 'flow' for a bond in the sensitivity bond graph represents a quasi-power quantity. There are two possibilities: large-scale parameter changes and small-scale parameter changes. The latter case is discussed in this paper, and is characterized by evaluation of the derivatives of the effort and flow functions with respect to the parameters of the elements, including sources of effort and flow. The variations in effort and flow can be caused directly by changes in the parameters of the elements, or indirectly by external changes. For example, the values of R, C, and I elements can be functions of ambient temperature, humidity, or pressure, or they may themselves vary over time. The paper indicates the relationships between the parent bond graph and the sensitivity bond graph for a variety of cases, and gives simple practical examples.

Introduction

It has been amply shown that there is a direct and one-to-one correspondence between the bond graph model of any system, and the linear graph model of the same system[1,2,3,4,5]. Additionally, it has been known for a long time that sensitivities to small changes in parameters can be calculated using the sensitivity graph approach, in which these sensitivities are obtained from the solution of a linear graph model which is topologically identical to the original graph model of the system. The variables associated with the edges in the sensitivity graph are the partial derivatives of the variables in the original graph with respect to a particular parameter.

It follows immediately that there is a bond graph model that has one-to-one correspondence with the sensitivity linear graph, and that its solution will indeed be the same as that of the sensitivity graph model. Moreover, since the original linear graph and its sensitivity graph are topologically equivalent, so the original bond graph and its corresponding sensitivity bond graph are topological identical.

Sensitivity modeling

In this section we review the mathematical relations involved in calculating sensitivities, which are assumed to be the partial derivatives of variables with respect to parameters other than time. All physical systems are made up of components with specific characteristics.

Components

We consider only R, C and I elements, along with sources of effort and flow. For illustration, consider the electrical case. R elements are electrical resistances, C elements are capacitors, and I elements are inductors. sources of effort and flow are voltage and current sources, respectively. Clearly, the components are characterized by:

$v = Ri$	R element	(1)
$i = Csv - Cv(0)$	C element	(2)
$v = Isi - Ii(0)$	I element	(3)
$v = E$	Source of voltage (E is a known function of time)	(4)
$i = J$	Source of current (J is a known function of time)	(5)

Note that we have written these equations in the LaPlace domain, so as to distinguish clearly derivatives with respect to parameters as opposed to time.

Now assume that all of R, C, I, E, J can be functions of some parameter, p. p can be one of the elements itself, or some other external parameter. We find the partial derivatives of equations (1) to (5) with respect to p. For convenience, we use the notation:

$$\frac{\partial f}{\partial p} \equiv f^p \tag{6}$$

and we obtain, for example , from (1):

$$v^p = R^p i + R i^p \tag{7}$$

Similarly, for a voltage source, we obtain:

$$v^p = E^p \tag{8}$$

and similarly for a current source.

Now suppose, for simplicity, that the parameter , p, is exactly the kth R element, R_k. Then R^p in (7) is always zero, except for the kth element, and we have:

$$v_j^p = R_j i_j^p \tag{9}$$

for $j \neq k$, and

$$v_k^p = i_k + R_k i_k^p \tag{10}$$

Note that (10) has the same form as (3), in the sense that there is an additive term which in (2) is an initial condition, while in (10) is the value of a variable obtained from solution of the original system. For sources, in this simple case, note that $v^p = 0$ and $i^p = 0$.

It is worthwhile to point out that when the parameter, p, of interest is not one of the elements, there is no convenient, general reduction in equation (7) for any element, and all the elements will be characterized by equations like (10). The reader can easily fill in the details.

Additionally, if p is one of the source functions, E or J, all the other parameter equations are reduced to the type of (9). There will be one source of effort or flow, whose partial derivative leads to an equation like:

$$v_k^p = 1 \tag{11}$$

Constraints

Constraints are the relations implied by the junction structure of the bond graph of a system, or the topology of the linear graph of the system. They are therefore in the form of two sets of equations, one a set of linear combinations of efforts, the other a set of linear combinations of flows. They are thus of the form:

$$\sum_{k=1}^{n} b_k v_k = 0 \tag{12}$$

with the $b_k = \pm 1$ or 0. The sum is over all the bonds, or all the edges. For the flows, we have equations of the form:

$$\sum_{k=1}^{n} q_k i_k = 0 \tag{13}$$

with the $q_k = \pm 1$ or 0. Again the sum is over all the bonds in the system bond graph or all the edges in the system graph.

There is one equation of the type of (12) or (13) for every junction in the bond graph, and for every vertex (cut set) and circuit in the linear graph model.

We now differentiate (12) and (13) with respect to the chosen parameter, p. This yields:

$$\sum_{k=1}^{n} b_k v_k^p = 0 \tag{14}$$

$$\sum_{k=1}^{n} q_k i_k^p = 0 \tag{15}$$

Equations (14) and (15) are the same as (12) and (13), except that the efforts and flows have been replaced by their partial derivatives with respect to p. Thus, they can be taken to represent the junction structure of a (pseudo) bond graph which has these derivative functions as its efforts and flows.

On the other hand, we have a set of component relations which are altered only slightly from those of the original system. As seen above, all the relations for the R, C and I elements are unchanged save one, and this one is readily identified. The sources of efforts and flows are reduced either to zero or to 1.

The one element that has its characteristic changed now satisfies equation (10). Here, i_k is given by the solution of the original system under the assumption that no variations will take place. This becomes a known quantity in (10). Hence, The sensitivities of all the variables of the original system with respect to the parameter p can be calculated recursively using either the bond graph or the linear graph technique by the following process:

1. Find the solution to the original system, using bond graphs or linear graphs.
2. Replace the efforts and flows in the original bond graph by their derivatives with respect to the parameter, p.
3. Obtain a new linear graph, or a new bond graph. These have the same topology or junction structure as the originals.
4. Replace the component characteristics from the original R, C and I elements by those given in equations (7), (8), (9), (10) and the like.
5. Solve using this bond graph (or linear graph). This is recursive because the new I, C and R elements contain parameters found in the original system.

The bond graph in this process is the sensitivity bond graph of the original system, with respect to the parameter, p. Several things are noteworthy:

1. A different sensitivity bond graph appears for every parameter, p. In the event that one wishes to establish the sensitivity of the outputs of a system to changes in all its component parameters, there will be as many sensitivity bond graphs as there are external bonds in the original system.
2. The method is not restricted to bond graphs with R, C and I elements only. Gyrators, transformers, modulated transformers, etc., can be included, and the parameters of these components can be among those with respect to whose changes output sensitivities can be obtained.
3. The method is recursive. Second and higher order sensitivities can be found by obtaining the sensitivity bond graph of the sensitivity bond graph, and so on.
4. Since all the sensitivity bond graphs have the same junction structure, and since most of the components have unchanged characteristics, i.e., the external bonds are unaltered, the computation beyond the first, or original, level is considerably reduced. Space considerations will not permit us to furnish the details here.

Example

Consider the simple example of an electric circuit shown in Figure 1. The Bond graph is shown in Figure 2, while the Linear Graph is in Figure 3.

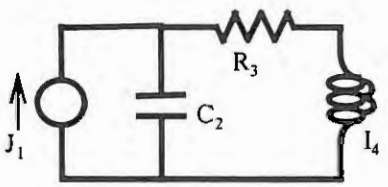


Figure 1. A Simple electric Circuit

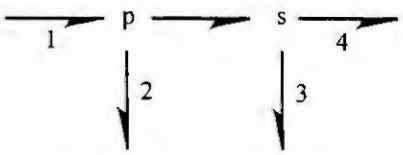


Figure 2. The bond graph corresponding to Fig. 1.

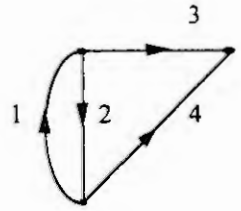


Figure 3. TheLinear Graph corresponding to Fig. 1.

The bond graph in Figure 2 and the linear graph in Figure 3 have the same structure as their sensitivity bond graphs and sensitivity graphs. If we wish to find the sensitivities of the currents and voltages in the system to changes in the resistance, R_3, the voltages and currents in the bond graph and linear graph models are replaced by their derivatives with respect to R_3, and the constituent relations for the components (the external bonds of Figure 2 and the edges of Figure 3), become:

$$i_1^{R_3} = 0 \tag{16}$$

$$i_2^{R_3} = C_2 s v_2^{R_3} \tag{17}$$

$$v_3^{R_3} = R_3 i_3^{R_3} + i_3 \tag{18}$$

$$v_4^{R_3} = I_4 s i_4^{R_3} \tag{19}$$

The solution proceeds by first finding $i3$ through normal bond graph or linear graph techniques[6,7,8], and using this in Equation (18). Then relations (16) through (19) replace the original component equations, and are solved for the first order sensitivities.

Conclusions

We have shown that there is a simple, recursive bond graph modeling method for obtaining first and higher order sensitivities of system variables with respect to any parameter variation. The method involves repeated solution of a given bond graph model with changing component constituent relationships for each order. It is mathematically equivalent to a linear graph technique that has been reported in the past.

References

1. S.H. Birkett & P.H. Roe The Mathematical Foundations of Bond Graphs -I-Algebraic Theory, J. Franklin Institute, 326 No. 3, pp.329-350,1989

2. Peter H. Roe & J.U. Thoma, Orientation and Causality in Bondgraphs and their Connection with Linear Graphs, , CESA '98, IMACS Multiconference, Nabeuil-Hammemet, Tunisia, 1998

3. S.H. Birkett & P.H. Roe, The Mathematical Foundations of Bond Graphs - III. - Matroid Theory, J. Franklin Inst., 327, No. 1, pp 87-108, 1990

4. S.H. Birkett & P.H. Roe, The Mathematical Foundations of Bond Graphs -II Duality, J. Franklin Inst., 326, No. 5 pp 691-708, 1989

5. Chandrashekar, P.H. Roe & G.J. Savage, Graph Theoretic Models, - A Unifying Modeling Approach, Proc. 23rd. Conf. on Modeling and Simulation, Pittsburgh, May 1992.

6. Jean U. Thoma, Simulation by Bondgraphs, Springer-Verlag, Berlin, 1990

7. Jean U. Thoma, Introduction to Bond Graphs and their Applications, Pergamon International Library of Science, Technology, Engineering and Social Studies, Paris, 1975.

8. P.H.Roe, Networks and Systems, Addison-WesleyPublihing Co., Reading, Mass., U.S.A., 1966

MULTI-INPUT MULTI-OUTPUT MODELS FOR AIRCRAFT FLIGHT CONTROL SYSTEM DESIGN

D.J. Murray-Smith
Centre for Systems and Control and Department of Electronics and Electrical Engineering
University of Glasgow
Glasgow G12 8QQ, Scotland, U.K.

Abstract. This paper addresses problems of developing robust reduced-order multivariable models of aircraft for cases in which strong inter-axis coupling exists. Attention is focused particularly upon helicopter modelling and the associated problems of helicopter flight control system design. In that particular case *a priori* information about parameters of the vehicle model, especially parameters associated with certain types of cross-coupling, is usually limited. System identification techniques, and frequency-domain methods in particular, offer potential benefits for helicopter model parameter estimation and for external validation of models. Frequency-domain methods also offer an effective approach to model reduction for multi-input multi-output models and for some aspects of control system design. The helicopter case is used to illustrate how frequency-domain tools can be combined to provide an integrated environment for both model development and control system design.

Introduction

Mathematical models used for aircraft flight control system design and for the development of real-time piloted flight simulators are usually obtained by the application of fundamental physical laws and the principles of aircraft flight mechanics. Such models are generally nonlinear in form and involve a number of different control surfaces and actuators (e.g. elevator, ailerons and rudder in fixed-wing aircraft or the main-rotor collective pitch, main-rotor longitudinal and lateral cyclic pitch and tail-rotor collective pitch in the case of conventional helicopters). There are also several different output variables which are of primary interest for flight control system design. Parameter values for these physically-derived models of aircraft may be obtained from theoretical analysis, from wind-tunnel testing or from flight test data [1].

For the purposes of control system design the underlying nonlinear model is usually linearised for selected flight conditions (e.g. forward speed, altitude etc.) and aircraft configuration (e.g. mass, position of the centre of gravity etc.). Following control system design based upon linearised descriptions the overall performance of the resulting system may be evaluated through simulation studies involving the full nonlinear aircraft model [2]. Modifications may be necessary in the control system following these simulation-based investigations.

This paper describes the development of an integrated set of frequency-domain tools for aircraft modelling and control purposes, including system identification and parameter estimation, the reduction of linearised multi-input multi-output models and flight control system design. Emphasis has been placed upon frequency-domain methods because of the physical insight which such approaches offer for modelling and the importance of the frequency-domain in the context of classical single-input single-output control system design techniques which still provide the basis for many practical flight control laws even when used as part of a parallel process which brings them together with multivariable techniques [3].

System Identification for Aircraft Model Development

System identification techniques have been used with considerable benefit for the development of aircraft models for use in the development of high bandwidth flight control systems [4] and for the external validation of flight mechanics models of both fixed-wing aircraft and rotorcraft [5]. The level of difficulty encountered in the practical application of system identification methods to helicopters is, however, recognised as being significantly greater than in the case of fixed-wing aircraft due to the high level of measurement noise and the more complex aeromechanics of these vehicles, particularly in terms of the strength of inter-axis coupling [6].

Experiment design for system identification is also more difficult in the case of helicopters due to the inherent instability of such vehicles in some flight conditions and the *a priori* model uncertainties.

In recent years increasing emphasis has been placed on the use of frequency-domain methods for aircraft system identification. A number of successful approaches have been developed which have been evaluated in flight with several different vehicles [7-10].

The flexibility of the frequency-domain approach in tailoring the structure and parameters of the identified model to a restricted range of frequencies is one attractive feature of these methods. A second useful feature is the ease with which time delays can be incorporated into relatively low-order descriptions to account both for actual physical delays and unmodelled higher order dynamics. These features have been shown to result in improved model fits and more physically realistic values of model parameters in cases where estimates obtained by more conventional time-domain methods are known to be adversely affected by rotor modes which are associated with features not included in the model [8]. Measures of accuracy and linearity of identified frequency responses which can be provided from coherence functions are also of considerable value in the interpretation of frequency-domain identification results [10].

The validation or empirical improvement of a theoretical nonlinear model derived using classical flight mechanics modelling methods can be approached by comparing a series of linearised descriptions derived from the nonlinear model for different flight conditions with equivalent empirical models obtained using system identification methods for the same selected flight conditions [11]. Theoretical linearised descriptions of this kind will normally be of the same order as the original nonlinear model and some form of model reduction is often needed in order to make meaningful comparisons with models obtained by system identification.

Model Reduction in the Frequency Domain

The estimation of the structure and parameters of an appropriate multi-input multi-output (MIMO) model which has dynamic characteristics which approximate those of a given model of higher order is a familiar problem which can be approached in a number of ways. One frequency-domain approach is based upon an extension of a complex curve-fitting method first published by Levy in 1959 [12] for the single-input single-output case (SISO) and subsequently refined by others [13,14].

In its original form Levy's method used a modified least-squares approach to fit a SISO transfer function to measured response data. The method was based upon minimisation of the sum of squares of errors between the absolute magnitude of the measured frequency response data and the corresponding quantity from the model for a range of frequency values. Such an approach is readily applicable to the model reduction problem where the frequency response of a given high order description is substituted for the measured response data.

Modifications subsequently introduced for the SISO case allow the technique to be used with greater confidence over a range of frequencies which covers several decades and allow the errors in chosen parts of the frequency range to be given particular emphasis [13,14]. More recently Levy's approach has been extended to the single-input multi-output (SIMO) and the MIMO cases by Gong and Murray-Smith [15]. Transfer functions of reduced order models of a given system in state space form obtained using this MIMO approach are constrained to have the same denominators since the poles of the various different transfer functions linking inputs to outputs must be the same and must relate to eigenvalues of the given system matrix.

The SIMO and MIMO versions of the Levy method have been applied with success to the development of reduced order models of an advanced fighter aircraft and a large four-engined passenger jet transport [15]. The approach has also been used successfully in deriving 8^{th} order reduced models of a Puma helicopter for a number of different flight conditions from equivalent state space descriptions of fourteenth order [16].

Although MIMO transfer function descriptions are useful for many purposes the traditional starting point for flight control system design involves a vehicle description in linear state space form. The frequency-domain reduction method outlined above provides results in the form of a set of transfer functions and it is useful to be able to translate this back to a reduced-order state space description. One approach to this problem is provided by R.T. Chen's nonlinear inverse formulation [17,18]. Chen's inversion procedure provides a unique description because the relationship between model states and the chosen outputs is known. Tischler [18], in a system identification context, has proposed use of an iterative procedure to resolve any physical inconsistencies arising from the application of Chen's inverion method and to determine an appropriate model structure.

Frequency-Domain Methods for Multivariable Control System Design

The highly coupled nature of helicopter dynamics introduces major difficulties in applying 'one-loop-at-a-time' control system design methods based on conventional SISO techniques. Recent research on helicopter flight control has concentrated largely on techniques such as H_∞ [19], eigenstructure assignment [20] and linear quadratic Gaussian/loop transfer recovery [21]. The difficulty in using such methods arises from the fact that they do not provide the level of physical insight and 'visibility' of the classical approaches used for SISO applications. A further difficulty with techniques such as H_∞ is that they can give rise to controllers of relatively high order.

The techniques of individual channel analysis and design (ICAD) which have been developed by Leithead and O'Reilly [22] as the basis of a general frequency-domain approach to multivariable control system analysis and design have been shown recently to offer potential for rotorcraft flight control system applications [23,24]. Dudgeon and Gribble [24] have demonstrated through their work that ICAD is a convenient approach to use to obtain multivariable control laws that meet the revised helicopter handling qualities requirements ADS-33D [25]. They have shown that ICAD is particularly suitable for the helicopter flight-control problem because many ADS-33D performance requirements are expressed through frequency domain measures and are presented in terms of the response of specific outputs to specific reference inputs which is central to the ICAD methodology.

An Integrated Computing Environment for Modelling and Control Investigations

Within the Centre for Systems and Control at the University of Glasgow a considerable amount of experience has been built up over the last decade in terms of helicopter modelling and control. More specifically a considerable amount of effort has gone into the development of frequency-domain techniques for system identification, for model reduction, and for flight control system analysis and design. Software tools have been developed in each of these areas.

Efforts are now being made to bring all the different elements together in the form of a properly integrated suite of software with a well-defined and carefully designed user interface. The chosen computing environment for this is based upon general purpose personal computers and MATLAB software.

Separate routines for identification, model reduction and controller design will not only be brought together but will also allow use of nonlinear simulation models in a convenient and user-friendly fashion. The choice of MATLAB as the software environment allows use to be made of other relevant specialist software, such as the MATLAB handling Qualities Toolbox developed by Howitt [26] and also ensures compatibility with research collaborators on other sites.

Conclusions

Frequency-domain techniques can provide insight which is of particular value for the development of models of helicopters and other aircraft which display strong inter-axis coupling. This is true both at the system identification stage and in the development of reduced order models. Frequency-domain techniques for the analysis and design of multivariable control systems also provide physical understanding and 'visibility' in the design process which is not so directly available in other multivariable design approaches. An integrated computing environment for modelling and flight control system design, based on frequency-domain methods, is a potentially useful development.

References

1. Tischler, M.B. (Editor), Advances in Aircraft Flight Control. Taylor and Francis, London, 1996.
2. Murray-Smith, D.J., Modelling limitations for helicopter flight control system design. In Breitenecker, F. and Husinsky, I. Proceedings 1995 EUROSIM Conference, EUROSIM '95, Vienna, Elsevier, Amsterdam, 1995, 397-402.

3. Moorhouse, D.J. and Citurs, K.D., Practical aspects of the design of an integrated flight and propulsion control system. In: Advances in Aircraft Flight Control, (Ed. Tischler, M.B.), Taylor and Francis, London, 1996, 369-412.
4. Hamel, P.G., Aerospace vehicle modelling requirements for high bandwidth flight control. In: Aerospace Vehicle Dynamics and Control (Eds. Cook, M.V. and Rycroft, M.J.), Clarendon Press, Oxford, 1994, 1-31.
5. Murray-Smith, D.J., Methods for the external validation of continuous system simulation models: a Review. Mathematical and Computer Modelling of Dynamical Systems, 4 (1998), 5-31.
6. Hamel, P.G. (Editor), Rotorcraft System Identification, AGARD-AR-280, AGARD, Neuilly sur Seine, France, 1991.
7. Fu, K.H. and Marchand, M., Helicopter system identification in the frequency domain. In: Proc. 9th European Rotorcraft Forum, Stresa, Italy, 1983, Paper 96.
8. Black, C.G. and Murray-Smith, D.J., A frequency-domain system identification approach to helicopter flight mechanics model validation. Vertica, 13 (1989), 343-368.
9. Tischler, M.B., Identification techniques: frequency domain methods. In: Rotorcraft System Identification, AGARD-LS-178, AGARD, Neuilly sur Seine, France, 1991.
10. Tischler, M.B. and Cauffman, M.G., Frequency-response method for rotorcraft system identification: flight applications to BO 105 coupled rotor/fuselage dynamics. J. American Helicopter Soc., 379 (1992), 3-17.
11. Bradley, R., Padfield, G.D., Murray-Smith, D.J. and Thomson, D.G., Validation of helicopter mathematical models, Trans. Institute of Measurements and Control, 12 (1990), 186-196.
12. Levy, E.C., Complex curve fitting. IRE Trans. on Automatic Control, AC-4 (1959), 37-44.
13. Sanathan, C.K. and Koerner, J., Transfer function synthesis as a ratio of two complex plynomials. IEEE Trans. on Automatic Control, AC-8 (1963), 56-58.
14. t'Mannetje, J.J., Transfer function identification using a complex curve-fitting technique. J. Mechanical Eng. Sci., 15 (1973), 339-345.
15. Gong, M. and Murray-Smith, D.J., Model reduction by an extended complex curve-fitting approach. Trans. Institute of Measurement and Control, 15 (1993), 188-198.
16. Gong, Mingrui, Model Reduction Techniques and their Application to Helicopter Models. M.Sc. Thesis, University of Glasgow, 1992.
17. Chen, R.T.N., Transfer-characterization and the unique realization of linear time-invariant multivariable systems. In: Proc. 1972 Joint Automatic Control Conf., Stanford, 1972, 238-247.
18. Tischler, M.B., Advancements in frequency-domain methods for rotorcraft system identification, Vertica, 13 (1989) 327-342.
19. Walker, D.J. and Postlethwaite, I., Advanced helicopter flight control using two-degree-of-fredom H infinity optimization. J. Guidance Control and Dynamics 19 (1996), 461-468.
20. Manness, M.A. and Murray-Smith, D.J., Aspects of multivariable flight control law design for helicopters using eigenstructure assignment, J. American Helicopter Soc. 37 (1992), 18-32.
21. Gribble, J.J., Linear-Quadratic Gaussian loop transfer recovery design for a helicopter in low-speed flight. J. Guidance Control and Dynamics, 16 (1993), 754-761.
22. Leithead, W.E. and O'Reilly, J., M-input m-output feedback control issues by individual channel design. Part 1, Structural issues. Int. J. Control, 56 (1992), 1347-1397.
23. Dudgeon, G.J.W., Gribble, J.J. and O'Reilly, J., Individual channel analysis and helicopter flight control in moderate- and large-amplitude manoeuvres. Control Eng. Practice, 5 (1997), 33-38.
24. Dudgeon, G.J.W. and Gribble, J.J., Helicopter translational rate command using individual channel analysis and design. Control Eng. Practice, 6 (1998), 15-23.
25. Anon., Handling qualities requirements for military rotorcraft (ADS-330). United States Army Aviation Systems and Troop Command, Directorate of Engineering, St. Louis, MO, U.S.A., 1994.
26. Howitt, J., MATLAB toolbox for handling qualities assessment of flight control laws. In: Proceedings IEE International Conference on Control (Control '91), IEE, London, 1991, 1251-1256.

MODELLING FOR MODULAR ANALYSIS AND DESIGN OF INTEGRATED SYSTEMS

D.J. Leith, W.E. Leithead
Department of Electronic & Electrical Engineering, University of Strathclyde,
50 George St., Glasgow G1 1QE, U.K.
Tel. +44 141 548 2407, Fax. +44 141 548 4203, Email. doug@icu.strath.ac.uk

Abstract
The selection of an appropriate representation for linearisation-based analysis and design of integrated systems is considered. It is shown that, in contrast to conventional linearisation-based representations, a velocity-based representation supports the modular analysis and design methodologies required with complex integrated systems.

1. Introduction

In response to increasingly stringent performance requirements, the trend in a wide range of engineering systems is towards tighter integration of the elements constituting the system. Whilst this frequently leads to increased dynamic interaction between sub-systems, the design of a complex system in an efficient and flexible manner necessitates a modular methodology whereby each sub-system has a well-defined interface to the rest of the system which is insensitive to the implementation details of the system. Such an approach enables the detailed design and implementation of each sub-system to be carried out separately and this is particularly important in projects where sub-contractors are involved.

There is, of course, a corresponding requirement for a representation which supports the analysis and design of systems in a modular fashion. Depending on the task at hand, a number of different representations are typically utilised for the analysis and design of complex systems; for example, Bond Graphs, Ordinary Differential Equations and Transfer Functions. Each representation possesses particular advantages and disadvantages which make it more, or less, suitable for a particular purpose. Traditionally, linearisation-based representations are often employed for the analysis and design of nonlinear systems. The system is approximated by a suitable linear system which, in the vicinity of an equilibrium operating point, exhibits similar dynamic characteristics (typically, the series expansion linearisation about the equilibrium operating point). Whilst conventional linearisation-based analysis is only valid locally to a specific equilibrium operating point, it has the considerable advantage that it maintains continuity with established linear analysis techniques for which a substantial body of experience has been accumulated. In order to determine appropriate linear approximations to a particular sub-system, knowledge of the equilibrium operating points is required. However, specifying *a priori* the relationship between the equilibrium inputs and outputs of a sub-system imposes, in general, a very strong restriction on the characteristics of the sub-system. For example, consider the nonlinear system $\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x}, \mathbf{r})$, $\mathbf{y} = \mathbf{G}(\mathbf{x}, \mathbf{r})$. The functions, $\mathbf{F}(\bullet,\bullet)$ and $\mathbf{G}(\bullet,\bullet)$, are not independent since, $\mathbf{F}(\mathbf{x}_o, \mathbf{r}_o) = 0$, $\mathbf{G}(\mathbf{x}_o,\mathbf{r}_o) = \mathbf{y}_o$ must be jointly satisfied for all equilibrium input-output pairs $(\mathbf{x}_o, \mathbf{r}_o)$. Indeed, when $\mathbf{F}(\bullet,\bullet)$ is invertible so that the equilibrium state, $\mathbf{x}_o$, is determined by $\mathbf{r}_o$, it follows that the output function, $\mathbf{G}(\bullet,\bullet)$, is, essentially, completely specified by the equilibrium input-output relationship and the choice of $\mathbf{F}(\bullet,\bullet)$. Consequently, in practice, a flexible equilibrium specification is required. However, since the equilibrium operating points are not those of the isolated sub-system but rather those of the sub-system when it is embedded in the overall system, the equilibrium operating points of a sub-system are, in general, strongly influenced by the characteristics of the overall integrated system and may change considerably as a result of even relatively small changes in other sub-systems. Hence, conventional linearisation-based representations do *not* readily support modular analysis and design approaches.

Whilst, in the context of integrated systems, the foregoing is perhaps the primary deficiency of conventional linearisation-based representations, it should be noted that there are also a number of other difficulties with such techniques. Firstly, the representations are only accurate in the vicinity of an equilibrium operating point whilst the requirement is usually to design a system which functions well not only when operating in the vicinity of a single equilibrium point but also during transitions between equilibrium operating points and periods of sustained non-equilibrium operation. Conventionally, this requirement is addressed by employing extensive simulation studies to iteratively refine the design, but this quickly becomes extremely time-consuming and inefficient for any but the simplest nonlinear systems. There is, therefore, a considerable incentive to directly incorporate, into the analytical part of the design procedure, knowledge of the plant dynamics during transitions between equilibrium operating

points and during sustained non-equilibrium operation. Secondly, a number of distinct linear representations are typically employed during analysis and design (Leith & Leithead 1998a) which are not equivalent and which can make it difficult to incorporate insight obtained from the analysis into the design procedure. Thirdly, and at a more practical level, the determination of the equilibrium operating point is time-consuming and highly non-trivial for a complex nonlinear system. Similarly, the numerical differentiation associated with conventional numerical linearisation about an equilibrium point is an undesirable ill-conditioned operation.

The velocity-based representation recently proposed by Leith & Leithead (1998a) rigorously generalises and extends the conventional series expansion linearisation at an equilibrium operating point and associates a linear system with every operating point, not just equilibrium operating points. An alternative description of a nonlinear system in terms of a family of linear systems is thereby established; namely, the velocity-based linearisation family. It is emphasised that the velocity-based formulation involves no loss of information and, in particular, does not involve either an inherent slow variation restriction nor is it confined to equilibrium operating points alone but instead encompasses every operating point, including those far from equilibrium. The velocity-based approach thereby relaxes the restriction to near equilibrium operation whilst maintaining the continuity with linear methods which is a principle advantage of the conventional linearisation-based analysis techniques. The velocity-based representation therefore resolves many of the deficiencies of conventional linearisation-based representations. However, the literature on the velocity-based representation is confined to "monolithic" systems for which there is no requirement to consider a decomposition into component sub-systems. The aim of this paper is, therefore, to investigate the application of the velocity-based representation to integrated systems and, in particular, consider the support, if any, provided by this representation for modular analysis and design.

2. Velocity-based representation

Before considering integrated systems, the velocity-based analysis and design representation is briefly summarised. Consider a nonlinear system

$$\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x}, \mathbf{r}), \quad \mathbf{y} = \mathbf{G}(\mathbf{x}, \mathbf{r}) \tag{1}$$

where $\mathbf{F}(\cdot,\cdot)$ and $\mathbf{G}(\cdot,\cdot)$ are differentiable nonlinear functions and $\mathbf{r} \in \Re^m$ denotes the input to the plant, $\mathbf{y} \in \Re^p$ the output and $\mathbf{x} \in \Re^n$ the states. It can be shown (Leith & Leithead 1998a) that the solution $\hat{\mathbf{x}}$ to the linear system (the "velocity-based linearisation")

$$\dot{\hat{\mathbf{x}}} = \hat{\mathbf{w}}\,, \dot{\hat{\mathbf{w}}} = \nabla_{\mathbf{x}}\mathbf{F}(\mathbf{x}_1, \mathbf{r}_1)\,\hat{\mathbf{w}} + \nabla_{\mathbf{r}}\mathbf{F}(\mathbf{x}_1, \mathbf{r}_1)\,\dot{\mathbf{r}} \tag{2}$$

$$\dot{\hat{\mathbf{y}}} = \nabla_{\mathbf{x}}\mathbf{G}(\mathbf{x}_1, \mathbf{r}_1)\,\hat{\mathbf{w}} + \nabla_{\mathbf{r}}\mathbf{G}(\mathbf{x}_1, \mathbf{r}_1)\,\dot{\mathbf{r}} \tag{3}$$

approximates the solution $\mathbf{x}$ to the nonlinear system locally to the operating point $(\mathbf{x}_1, \mathbf{r}_1)$. Since a linear system (2)-(3) is associated with every operating point of the nonlinear system, there is a family of velocity-based linearisations associated with the nonlinear system. Whilst the solution to a single velocity-based linearisation is only a local approximation to the solution of the nonlinear system, the solutions to the members of this family can be pieced together to obtain an arbitrarily accurately *global* approximation to the solution of the nonlinear system. Indeed, the family of velocity-based linearisations embodies the entire dynamics of the nonlinear system, (1), with no loss of information and provides an alternative representation of the nonlinear system.

3. Integrated systems

The velocity-based representation resolves many of the deficiencies of conventional linearisation-based analysis and design techniques. However, the existing literature on the velocity-based representation is confined to "monolithic" systems for which there is no requirement to consider a decomposition into component sub-systems. With regard to integrated systems, the velocity-based representation, in contrast to conventional linearisation-based representations, is not restricted to near equilibrium operation and does *not* require the equilibrium operating point of a system to be determined before linearisation-based analysis is possible. Rather, application of velocity-based analysis and design techniques only involves the much weaker requirement that the largest operating envelope of a system is known. As noted previously, it is non-trivial to determine the equilibrium operating point of a sub-system when it is embedded in the overall system and, furthermore, the equilibrium operating point is, in general, strongly influenced by the characteristics of the overall integrated system. Since the requirement to determine an equilibrium operating point is relaxed in the velocity-based framework, this framework resolves one of the principal difficulties, in the context of integrated systems, with conventional linearisation-based techniques. Of course, whilst this is necessary in order to de-couple the analysis and design of the sub-systems, it is not sufficient to support modular analysis and design. It is, in addition, also required that the analysis and design results obtained with a specific sub-

system can be integrated in a direct and transparent manner with those obtained for other sub-systems. In order to investigate the integration of analysis and design results obtained for different sub-systems, it is sufficient to consider the series, parallel and feedback combination of sub-systems since these are the principal classes of interconnection in widespread use.

3.1 Series combination

Consider the nonlinear system $\dot{\mathbf{x}}_1 = \mathbf{F}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1), \quad \mathbf{y}_1 = \mathbf{G}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1)$ (4)

for which the velocity-based form is

$$\dot{\mathbf{x}}_1 = \mathbf{w}_1, \quad \dot{\mathbf{w}}_1 = \nabla_{\mathbf{x}_1}\mathbf{F}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1)\mathbf{w}_1 + \nabla_{\mathbf{x}_1}\mathbf{F}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1)\dot{\mathbf{z}}_1 + \nabla_{\mathbf{r}_1}\mathbf{F}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1)\dot{\mathbf{r}}_1 \quad (5)$$

$$\dot{\mathbf{y}}_1 = \nabla_{\mathbf{x}_1}\mathbf{G}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1)\mathbf{w}_1 + \nabla_{\mathbf{x}_1}\mathbf{G}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1)\dot{\mathbf{z}}_1 + \nabla_{\mathbf{r}_1}\mathbf{G}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1)\dot{\mathbf{r}}_1$$

and the nonlinear system $\dot{\mathbf{x}}_2 = \mathbf{F}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2), \quad \mathbf{y}_2 = \mathbf{G}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)$ (6)

for which the velocity-based form is

$$\dot{\mathbf{x}}_2 = \mathbf{w}_2, \quad \dot{\mathbf{w}}_2 = \nabla_{\mathbf{x}_2}\mathbf{F}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\mathbf{w}_2 + \nabla_{\mathbf{x}_2}\mathbf{F}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\dot{\mathbf{z}}_2 + \nabla_{\mathbf{r}_2}\mathbf{F}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\dot{\mathbf{r}}_2 \quad (7)$$

$$\dot{\mathbf{y}}_2 = \nabla_{\mathbf{x}_2}\mathbf{G}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\mathbf{w}_2 + \nabla_{\mathbf{x}_2}\mathbf{G}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\dot{\mathbf{z}}_2 + \nabla_{\mathbf{r}_2}\mathbf{G}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\dot{\mathbf{r}}_2$$

The systems, (4) and (6), are cascaded together by setting $\mathbf{z}_2=\mathbf{y}_1$. The resulting system is

$$\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x},\mathbf{r},\mathbf{z}), \quad \mathbf{y} = \mathbf{G}(\mathbf{x},\mathbf{r},\mathbf{z}) \quad (8)$$

where $\mathbf{x}=\begin{bmatrix}\mathbf{x}_1\\ \mathbf{x}_2\end{bmatrix}, \mathbf{r}=\begin{bmatrix}\mathbf{r}_1\\ \mathbf{r}_2\end{bmatrix}, \mathbf{z}=\mathbf{z}_1, \mathbf{y}=\mathbf{y}_2, \mathbf{F}(\mathbf{x},\mathbf{r},\mathbf{z})=\begin{bmatrix}\mathbf{F}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1)\\ \mathbf{F}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{G}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1))\end{bmatrix}, \mathbf{G}(\mathbf{x},\mathbf{r},\mathbf{z})=\mathbf{G}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{G}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1))$, for which the velocity-based form is

$$\dot{\mathbf{x}}=\mathbf{w}=\begin{bmatrix}\mathbf{w}_1\\ \mathbf{w}_2\end{bmatrix}, \dot{\mathbf{w}}=\begin{bmatrix}\dot{\mathbf{w}}_1\\ \dot{\mathbf{w}}_2\end{bmatrix}=\begin{bmatrix}\nabla_{\mathbf{x}_1}\mathbf{F}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1) & 0\\ \nabla_{\mathbf{x}_2}\mathbf{F}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\nabla_{\mathbf{x}_1}\mathbf{G}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1) & \nabla_{\mathbf{x}_2}\mathbf{F}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\end{bmatrix}\begin{bmatrix}\mathbf{w}_1\\ \mathbf{w}_2\end{bmatrix}+\begin{bmatrix}\nabla_{\mathbf{z}_1}\mathbf{F}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1)\\ \nabla_{\mathbf{x}_2}\mathbf{F}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\nabla_{\mathbf{z}_1}\mathbf{G}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1)\end{bmatrix}\dot{\mathbf{z}}_1+\begin{bmatrix}\nabla_{\mathbf{r}_1}\mathbf{F}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1) & 0\\ \nabla_{\mathbf{x}_2}\mathbf{F}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\nabla_{\mathbf{r}_1}\mathbf{G}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1) & \nabla_{\mathbf{r}_2}\mathbf{F}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\end{bmatrix}\begin{bmatrix}\dot{\mathbf{r}}_1\\ \dot{\mathbf{r}}_2\end{bmatrix}$$

$$\dot{\mathbf{y}}=\dot{\mathbf{y}}_2=\begin{bmatrix}\nabla_{\mathbf{x}_2}\mathbf{G}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\nabla_{\mathbf{x}_1}\mathbf{G}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1) & \nabla_{\mathbf{x}_2}\mathbf{G}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\end{bmatrix}\begin{bmatrix}\mathbf{w}_1\\ \mathbf{w}_2\end{bmatrix}+\nabla_{\mathbf{z}_1}\mathbf{G}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\nabla_{\mathbf{z}_1}\mathbf{G}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1)\dot{\mathbf{z}}_1+\begin{bmatrix}\nabla_{\mathbf{r}_1}\mathbf{G}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\nabla_{\mathbf{r}_1}\mathbf{G}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1) & \nabla_{\mathbf{r}_2}\mathbf{G}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\end{bmatrix}\begin{bmatrix}\dot{\mathbf{r}}_1\\ \dot{\mathbf{r}}_2\end{bmatrix} \quad (9)$$

$$\mathbf{z}_2=\mathbf{y}_1=\mathbf{G}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1)$$

Evidently, (9) is just the system obtained when the systems, (5) and (7), are cascaded together. It follows that the velocity-based form of a system consisting of two cascaded sub-systems is identical to the system obtained by cascading together the velocity-based forms of the two sub-systems.

3.2 Parallel combination

The systems, (4) and (6), are combined in parallel by setting $\mathbf{z}_2=\mathbf{z}_1$. The resulting system is

$$\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x},\mathbf{r},\mathbf{z}), \quad \mathbf{y} = \mathbf{G}(\mathbf{x},\mathbf{r},\mathbf{z}) \quad (10)$$

where $\mathbf{x}=\begin{bmatrix}\mathbf{x}_1\\ \mathbf{x}_2\end{bmatrix}, \mathbf{r}=\begin{bmatrix}\mathbf{r}_1\\ \mathbf{r}_2\end{bmatrix}, \mathbf{z}=\mathbf{z}_1, \mathbf{y}=\begin{bmatrix}\mathbf{y}_1\\ \mathbf{y}_2\end{bmatrix}, \mathbf{F}(\mathbf{x},\mathbf{r},\mathbf{z})=\begin{bmatrix}\mathbf{F}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1)\\ \mathbf{F}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_1)\end{bmatrix}, \mathbf{G}(\mathbf{x},\mathbf{r},\mathbf{z})=\begin{bmatrix}\mathbf{G}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1)\\ \mathbf{G}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_1)\end{bmatrix}$ (11)

for which the velocity-based form is

$$\dot{\mathbf{x}}=\mathbf{w}=\begin{bmatrix}\mathbf{w}_1\\ \mathbf{w}_2\end{bmatrix}, \dot{\mathbf{w}}=\begin{bmatrix}\dot{\mathbf{w}}_1\\ \dot{\mathbf{w}}_2\end{bmatrix}=\begin{bmatrix}\nabla_{\mathbf{x}_1}\mathbf{F}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1) & 0\\ \nabla_{\mathbf{x}_2}\mathbf{F}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\nabla_{\mathbf{x}_1}\mathbf{G}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1) & \nabla_{\mathbf{x}_2}\mathbf{F}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\end{bmatrix}\begin{bmatrix}\mathbf{w}_1\\ \mathbf{w}_2\end{bmatrix}+\begin{bmatrix}\nabla_{\mathbf{z}_1}\mathbf{F}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1)\\ \nabla_{\mathbf{x}_2}\mathbf{F}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\nabla_{\mathbf{z}_1}\mathbf{G}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1)\end{bmatrix}\dot{\mathbf{z}}_1+\begin{bmatrix}\nabla_{\mathbf{r}_1}\mathbf{F}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1) & 0\\ 0 & \nabla_{\mathbf{r}_2}\mathbf{F}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_1)\end{bmatrix}\begin{bmatrix}\dot{\mathbf{r}}_1\\ \dot{\mathbf{r}}_2\end{bmatrix} \quad (12)$$

$$\dot{\mathbf{y}}=\begin{bmatrix}\dot{\mathbf{y}}_1\\ \dot{\mathbf{y}}_2\end{bmatrix}=\begin{bmatrix}\nabla_{\mathbf{x}_1}\mathbf{G}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1) & 0\\ 0 & \nabla_{\mathbf{x}_2}\mathbf{G}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\end{bmatrix}\begin{bmatrix}\mathbf{w}_1\\ \mathbf{w}_2\end{bmatrix}+\begin{bmatrix}\nabla_{\mathbf{z}_1}\mathbf{G}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1)\\ \nabla_{\mathbf{z}_2}\mathbf{G}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\end{bmatrix}\dot{\mathbf{z}}_1+\begin{bmatrix}\nabla_{\mathbf{r}_1}\mathbf{G}_1(\mathbf{x}_1,\mathbf{r}_1,\mathbf{z}_1) & 0\\ 0 & \nabla_{\mathbf{r}_2}\mathbf{G}_2(\mathbf{x}_2,\mathbf{r}_2,\mathbf{z}_2)\end{bmatrix}\begin{bmatrix}\dot{\mathbf{r}}_1\\ \dot{\mathbf{r}}_2\end{bmatrix}$$

Evidently, (12) is just the system obtained when the systems, (5) and (7), are combined in parallel. Hence, the velocity-based form of a system consisting of the parallel combination of two sub-systems is identical to the system obtained by combining in parallel the velocity-based forms of the two sub-systems.

3.3 Feedback combination

Consider the nonlinear system with inputs, $\mathbf{r}$ and $\mathbf{z}$,

$$\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x},\mathbf{r},\mathbf{z}), \quad \mathbf{y} = \mathbf{G}(\mathbf{x},\mathbf{r},\mathbf{z}) \quad (13)$$

for which the corresponding velocity-based form is

$$\dot{\mathbf{x}} = \mathbf{w}, \quad \dot{\mathbf{w}} = \nabla_{\mathbf{x}}\mathbf{F}(\mathbf{x},\mathbf{r},\mathbf{z})\mathbf{w} + \nabla_{\mathbf{z}}\mathbf{F}(\mathbf{x},\mathbf{r},\mathbf{z})\dot{\mathbf{z}} + \nabla_{\mathbf{r}}\mathbf{F}(\mathbf{x},\mathbf{r},\mathbf{z})\dot{\mathbf{r}} \quad (14)$$

$$\dot{\mathbf{y}} = \nabla_{\mathbf{x}}\mathbf{G}(\mathbf{x},\mathbf{r},\mathbf{z})\mathbf{w} + \nabla_{\mathbf{z}}\mathbf{G}(\mathbf{x},\mathbf{r},\mathbf{z})\,\dot{\mathbf{z}} + \nabla_{\mathbf{r}}\mathbf{G}(\mathbf{x},\mathbf{r},\mathbf{z})\,\dot{\mathbf{r}}$$

Assuming that $\mathbf{y}=\mathbf{G}(\mathbf{x},\mathbf{r},\mathbf{y})$ (15)

has a suitable solution $\mathbf{y}=\mathbf{N}(\mathbf{x},\mathbf{r})$ (16)

the system, (13), is enclosed in a feedback loop by setting $\mathbf{z}=\mathbf{y}$. The resulting closed-loop system is

$$\dot{\mathbf{x}} = \mathbf{M}(\mathbf{x},\mathbf{r}), \quad \mathbf{y} = \mathbf{N}(\mathbf{x},\mathbf{r}) \tag{17}$$

with $\mathbf{M}(\mathbf{x},\mathbf{r}) = \mathbf{F}(\mathbf{x},\mathbf{r},\mathbf{N}(\mathbf{x},\mathbf{r}))$. The velocity-based form of (17) is

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{w}, \ \dot{\mathbf{w}} = \nabla_{\mathbf{x}}\mathbf{M}(\mathbf{x},\mathbf{r})\mathbf{w} + \nabla_{\mathbf{r}}\mathbf{M}(\mathbf{x},\mathbf{r})\,\dot{\mathbf{r}} \\ \dot{\mathbf{y}} &= \nabla_{\mathbf{x}}\mathbf{N}(\mathbf{x},\mathbf{r})\mathbf{w} + \nabla_{\mathbf{r}}\mathbf{N}(\mathbf{x},\mathbf{r})\,\dot{\mathbf{r}} \end{aligned} \tag{18}$$

Combining (15) and (16), $\mathbf{N}(\mathbf{x},\mathbf{r}) = \mathbf{G}(\mathbf{x},\mathbf{r},\mathbf{N}(\mathbf{x},\mathbf{r}))$ (19)

Hence,

$$\begin{aligned} &\nabla_{\mathbf{x}}\mathbf{M}(\mathbf{x},\mathbf{r}) = \nabla_{\mathbf{x}}\mathbf{F}(\mathbf{x},\mathbf{r},\mathbf{N}(\mathbf{x},\mathbf{r})) + \nabla_{\mathbf{z}}\mathbf{F}(\mathbf{x},\mathbf{r},\mathbf{N}(\mathbf{x},\mathbf{r}))\nabla_{\mathbf{x}}\mathbf{N}(\mathbf{x},\mathbf{r}), \nabla_{\mathbf{r}}\mathbf{M}(\mathbf{x},\mathbf{r}) = \nabla_{\mathbf{r}}\mathbf{F}(\mathbf{x},\mathbf{r},\mathbf{N}(\mathbf{x},\mathbf{r})) + \nabla_{\mathbf{z}}\mathbf{F}(\mathbf{x},\mathbf{r},\mathbf{N}(\mathbf{x},\mathbf{r}))\nabla_{\mathbf{r}}\mathbf{N}(\mathbf{x},\mathbf{r}) \\ &\nabla_{\mathbf{x}}\mathbf{N}(\mathbf{x},\mathbf{r}) = \nabla_{\mathbf{x}}\mathbf{G}(\mathbf{x},\mathbf{r},\mathbf{N}(\mathbf{x},\mathbf{r})) + \nabla_{\mathbf{z}}\mathbf{G}(\mathbf{x},\mathbf{r},\mathbf{N}(\mathbf{x},\mathbf{r}))\nabla_{\mathbf{x}}\mathbf{N}(\mathbf{x},\mathbf{r}), \nabla_{\mathbf{r}}\mathbf{N}(\mathbf{x},\mathbf{r}) = \nabla_{\mathbf{r}}\mathbf{G}(\mathbf{x},\mathbf{r},\mathbf{N}(\mathbf{x},\mathbf{r})) + \nabla_{\mathbf{z}}\mathbf{G}(\mathbf{x},\mathbf{r},\mathbf{N}(\mathbf{x},\mathbf{r}))\nabla_{\mathbf{r}}\mathbf{N}(\mathbf{x},\mathbf{r}) \end{aligned} \tag{20}$$

and, by substituting (20) into(18), the closed-loop system, (17), can be directly reformulated as

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{w}, \ \dot{\mathbf{w}} = \nabla_{\mathbf{x}}\mathbf{F}(\mathbf{x},\mathbf{r},\mathbf{z})\mathbf{w} + \nabla_{\mathbf{z}}\mathbf{F}(\mathbf{x},\mathbf{r},\mathbf{z})\,\dot{\mathbf{z}} + \nabla_{\mathbf{r}}\mathbf{F}(\mathbf{x},\mathbf{r},\mathbf{z})\,\dot{\mathbf{r}} \\ \dot{\mathbf{y}} &= \nabla_{\mathbf{x}}\mathbf{G}(\mathbf{x},\mathbf{r},\mathbf{z})\mathbf{w} + \nabla_{\mathbf{z}}\mathbf{G}(\mathbf{x},\mathbf{r},\mathbf{z})\,\dot{\mathbf{z}} + \nabla_{\mathbf{r}}\mathbf{G}(\mathbf{x},\mathbf{r},\mathbf{z})\,\dot{\mathbf{r}}, \ \mathbf{z} = \mathbf{y} = \mathbf{N}(\mathbf{x},\mathbf{r}) \end{aligned} \tag{21}$$

Since $\mathbf{N}(\mathbf{x},\mathbf{r})$ satisfies (19), it is clear that (21) is the system obtained when the system, (14), is enclosed in a feedback loop by setting $\mathbf{z}=\mathbf{y}$. Consequently, the velocity-based form of the closed-loop system is identical to the system obtained by enclosing the velocity-based form of the open-loop system in a feedback loop.

4. Conclusions

It is shown that the velocity-based representation supports modular linearisation-based analysis and design of integrated systems In particular, and in contrast to conventional linearisation-based representations, the velocity-based representation

- does not require an equilibrium operating point to be determined in order to analyse a system; rather, application of velocity-based analysis and design techniques only involves the much weaker requirement that the largest operating envelope of a system is known. Trimming, which is highly non-trivial for complex nonlinear systems, is not required.
- does not require numerical differentiation; rather the velocity-based linearisation is obtained by simply "freezing" the velocity form of the nonlinear system.
- is not confined to near equilibrium operation but rather accommodates both transitions between equilibrium operating points and sustained non-equilibrium operation.
- provides a unified framework for analysis and design which employs a single linearisation, namely the velocity-based linearisation.

Furthermore, the velocity-based representations of the series, parallel and feedback combination of two nonlinear systems are identical to the series, parallel and feedback combination of the velocity-based representations of the individual nonlinear systems. Hence, in addition to de-coupling the analysis/design of sub-systems, analysis and design results utilising the velocity-based representation of a specific sub-system can be integrated in a direct and transparent manner with those obtained for other sub-systems. The velocity-based representation therefore resolves many of the difficulties associated with conventional linearisation-based representations and provides direct support for the modular analysis and design methodologies required with complex integrated systems.

Acknowledgement

D.J.Leith gratefully acknowledges the support provided by the Royal Society for the work presented.

References

LEITH, D.J., LEITHEAD, W.E., 1998a, Gain-Scheduled & Nonlinear & Systems: Dynamic Analysis by Velocity-Based Linearisation Families. *Int. J. Control*, **70**, pp289-317; 1998b, Gain-Scheduled Controller Design: An Analytic Framework Directly Incorporating Non-Equilibrium Plant Dynamics. *ibid*, **70**, pp249-269; 1999a, Input-Output Linearisation by Velocity-based Gain-Scheduling. *ibid*, **72**, 229-246; 1999b, Analytic Framework for Blended Multiple Model Systems Using Linear Local Models. *ibid*, **72**, 605-619.

SUPERVISION OF SLAB REHEATING PROCESS USING MATHEMATICAL MODEL

A. Jaklič[1], T. Kolenko[2] and B. Glogovac[1]
[1]Institute of Metals and Technology
Lepi Pot 11, 1001 Ljubljana, Slovenia
[2]University of Ljubljana, Faculty of Natural Sciences and Technology
Aškerčeva 12, 1000 Ljubljana, Slovenia

Abstract. In the paper a mathematical model of slab reheating in gas heated pusher furnace is presented. Heat transfer inside the slab is calculated by finite difference method in two space dimensions. The heat exchange between the charge, the furnace gas, and the furnace wall is described by three temperature model. The model was implemented in industrial network system for real-time operation.

Introduction.

Modern computer automated hot rolling mill systems need quality reheated charge. Quality reheating means both that every slab satisfies optimum temperature trajectory as it is moved along the furnace and that the correct slab temperature for hot rolling dependent on steel grade as well as the minimum temperature difference in the slab is achieved at discharge from the furnace. A pusher furnace shown in Figure 1a can be used for the slab reheating. The material flow through the furnace is carried out by the pusher at regular intervals. However delay conditions in the material flow are often present. There are scheduled delays e.g. mill roll changes, meal breakes, etc. and unscheduled delays e.g. mill breakdowns, inadequate steel discharge, etc. All delays significantly affect the reheating process and demand appropriate zone temperature set points before, after and during these periods. Another influence on the reheating process have the slabs of different dimensions and steel grades that can simultaneusely reside in the furnace and can demand even various discharge temperature requirements. Standard procedures and rules for furnace temperature settings may be difficult to apply when described complex heating schedules are encountered and do not provide the furnace operator with the information needed to achieve quality reheating. By use of computer–based control system the operator is equiped with a valuable helping and decisioning tool.

The first computer control systems for slab reheating appeared in the early seventies.[4],[5] They were based on the mathematical model of heat conduction in the slabs and measurements of their surface temperatures. Improvements followed that rendered models to perform supervisory and regulatory control by combining feed–forward and feed–back algorithm.[2] However, computers of that time could not cope with large off–line models to be implemented for on–line control. Therefore, relationships were derived from off–line models which formed the basis for on–line computer control.[9],[6],[1] These simple computer aided systems provided instructions on desired furnace settings during normal throughput and delay conditions.

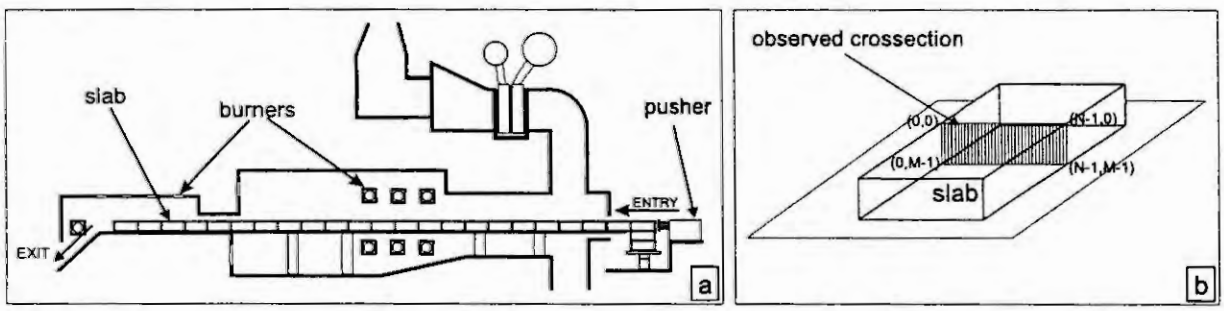


Figure 1: a) Pusher furnace; b) Observed slab crossection

Discussion of relative advantages and disadvantages of the different types of control strategies recommended a type of control strategy which matematically models the heat transfer process within the furnace from the first principles and is capable of tracking material through the furnace and on–line simulation of the thermal state of each slab.[10] By modern personal computers with their huge memory and computing capacities this approach is feasible. The sophisticated control system has been developed to bring the actual slab temperature on–line to the optimized reference trajectory.[11] The monitoring and control system can be connected to lower and higher level of informatization process via the plant Ethernet local area network (LAN).[7]

The supervision system for pusher furnace shown in Figure 1a in Acroni steelwork in Jesenice, Slovenia has been developed. For real-time operation the supervision system was implemented in industrial network system. The supervision system provides calculations of temperature field of each slab residing in the furnace as well as the calculation of heat balance, specific heat consumption and instantaneous thermal efficiency of the reheating process. The calculations mentioned base on the mathematical model which includes main physical phenomena appearing during the reheating process in the pusher furnace.

Modelling

In the model the cross section of the slab shown in Figure 1b is chosen, which is the compromise between the information of reheating quality and the calculation time. The observed cross section is of rectangular shape; therefore a finite difference method can be applied to solve heat transfer partial differential equation (1).

$$\rho c_p(\vartheta)\frac{\partial \vartheta}{\partial t} = \frac{\partial}{\partial x}\left(\lambda(\vartheta)\frac{\partial \vartheta}{\partial x}\right) + \frac{\partial}{\partial y}\left(\lambda(\vartheta)\frac{\partial \vartheta}{\partial y}\right) \tag{1}$$

The cross section is divided by rectangular mesh with M elements in x direction and with N elements in y direction. Their width is Δx and height is Δy. For each small element energy balance is derived under assumption of homogenious temperature field in the element. In the paper just three different types of elements are presented: the inner elements (Figure 2a and Equation (2)), the upper edge elements (Figure 2b and Equation (3)) and the upper left angle element (Figure 2c and Equation (4)). Equations for other elements can easily be derived by analogy of these three types of elements. In equation (2) denotes a thermal difusivity, λ thermal conductivity, i index on x axis (i is integer on interval $[0, N-1]$), j index on y axis (j is integer on interval $[0, M-1]$), ϑ temperature of element located dependent on indexes i and j. The recursive equation (2) enables calculation of temperature $\vartheta_{i,j}^{t+\Delta t}$ of inner element i, j at time $t + \Delta t$ by use of known temperatures: $\vartheta_{i-1,j}^{t}$, $\vartheta_{i+1,j}^{t}$, $\vartheta_{i,j-1}^{t}$ and $\vartheta_{i,j+1}^{t}$ of neighboring elements at time t. The recursive equations as e.g. (3) and (4) of outer elements have additional thermal fluxes e.g. $\dot{q}_{le}$ from left and $\dot{q}_{up}$ from upper side dependent on the position in the cross section. This outside thermal fluxes represent heat exchange between the observed cross section and the furnace enviroment; their unit is (W/m^2).

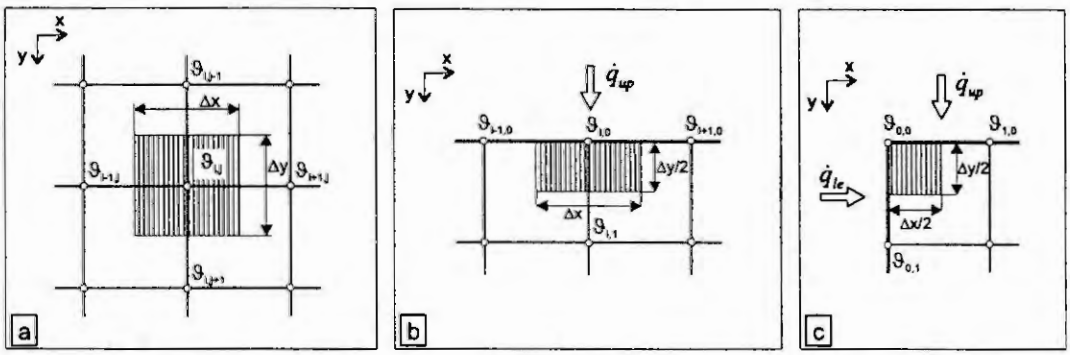


Figure 2: a) Inner elements; b) Upper edge elements; Upper left angle element

$$\vartheta_{i,j}^{t+\Delta t} = \frac{a\Delta t}{\Delta x \Delta y}\left[\frac{\Delta y}{\Delta x}\vartheta_{i+1,j}^{t} + \frac{\Delta y}{\Delta x}\vartheta_{i-1,j}^{t} + \frac{\Delta x}{\Delta y}\vartheta_{i,j+1}^{t} + \frac{\Delta x}{\Delta y}\vartheta_{i,j-1}^{t}\right] + \left[1 - \frac{2a\Delta t}{\Delta x \Delta y}\left(\frac{\Delta y}{\Delta x} + \frac{\Delta x}{\Delta y}\right)\right]\vartheta_{i,j}^{t} \tag{2}$$

$$\vartheta_{i,0}^{t+\Delta t} = \frac{a\Delta t}{\Delta x \Delta y}\left[\frac{\Delta y}{\Delta x}\vartheta_{i-1,0}^{t} + \frac{\Delta y}{\Delta x}\vartheta_{i+1,0}^{t} + \frac{2\Delta x}{\Delta y}\vartheta_{i,1}^{t} \pm \frac{2\Delta x}{\lambda}\dot{q}_{up}\right] + \left[1 - \frac{2a\Delta t}{\Delta x \Delta y}\left(\frac{\Delta y}{\Delta x} + \frac{\Delta x}{\Delta y}\right)\right]\vartheta_{i,0}^{t} \tag{3}$$

$$\vartheta_{0,0}^{t+\Delta t} = \frac{2a\Delta t}{\Delta x \Delta y}\left[\frac{\Delta y}{\Delta x}\vartheta_{1,0}^{t} + \frac{\Delta x}{\Delta y}\vartheta_{0,1}^{t} \pm \frac{\Delta y}{\lambda}\dot{q}_{le} \pm \frac{\Delta x}{\lambda}\dot{q}_{up}\right] + \left[1 - \frac{2a\Delta t}{\Delta x \Delta y}\left(\frac{\Delta y}{\Delta x} + \frac{\Delta x}{\Delta y}\right)\right]\vartheta_{0,0}^{t} \tag{4}$$

To evaluate the heat exchange between the charge, the furnace gas and the furnace wall, basic algorithms [3] of Heiligenstædt are used. The length of furnace is divided into small sections which are equal to the slab widths.

For this small section of the furnace, homogenous temperature field and homogenous furnace gas temperature field is assumed. The total heat transfer to the charge can be estimated by evaluation of partial mechanisms of heat transfer which can be seen in Figure 3. The furnace gas, which has the highest temperature in the system, emits heat to the charge and to the furnace wall. The heat is transfered from furnace gas to charge by radiation $\dot{q}_{rad\ gc}(\vartheta_g, \vartheta_c)$ and convection $\dot{q}_{conv\ gc}(\vartheta_g, \vartheta_c)$, and also from furnace gas to wall by radiation $\dot{q}_{rad\ gw}(\vartheta_g, \vartheta_w)$ and convection $\dot{q}_{conv\ gw}(\vartheta_g, \vartheta_w)$. The furnace wall gets the heat from the furnace gas. Under the steady state condition a part of this heat $\dot{q}_{cond\ w}(\vartheta_w)$ is lost outside through the furnace wall, another part of it is emited to the charge by wall radiation $\dot{q}_{rad\ wc}(\vartheta_w, \vartheta_c)$ being partly $\dot{q}_{abs\ g}(\vartheta_g)$ absorbed in the furnace gas. The significant factors in heat flux calculations are: the gas temperature ϑ_g, the inner wall temperature ϑ_w and the charge surface temperature ϑ_c. If two of the three mentioned temperatures are known, the third can be calculated from the equlibrium expression on the inner furnace wall surface (5)[3]. In the expression (5) A represents the area of wall surface which surrounds the area of heated surface a.

$$\begin{aligned} 0 &= \dot{q}_{rad\ gw}(\vartheta_g, \vartheta_w) \cdot A + \dot{q}_{conv\ gw}(\vartheta_g, \vartheta_w) \cdot A - \dot{q}_{cond\ w}(\vartheta_w) \cdot A - \\ &- \dot{q}_{rad\ wc}(\vartheta_w, \vartheta_c) \cdot a + \dot{q}_{abs\ g}(\vartheta_g) \cdot a \end{aligned} \tag{5}$$

By the Heiligenstædt model inner furnace wall surface temperature was chosen to be unknown. If this heat

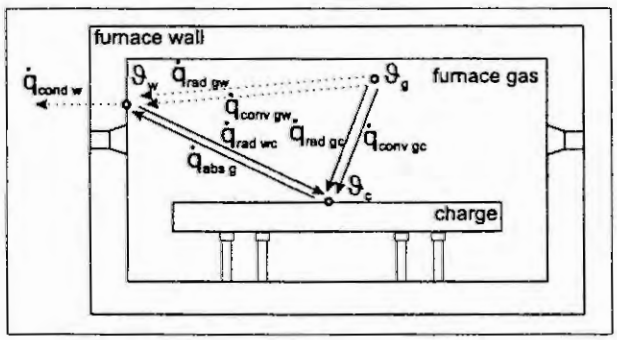


Figure 3: Heat exchange between the furnace gas, the furnace wall and the charge

transfer model is used for real-time aplications of industrial furnace monitoring, it turns out that it would be more suitable to treat the furnace gas temperature as the unknown variable [8], because the inner furnace wall surface temperature is more stable and can easier be measured than the furnace gas temperature. The charge surface temperature is also known since it is calculated step by step by finite difference method. The only unknown temperature is furnace gas temperature which can be calculated from expression (5). The unknown gas temperature ϑ_g can not be expressed explicitly from equation (5), therefore, it should be calculated numerically by finding ϑ_g at given ϑ_c and ϑ_w for which the expression (5) is fullfiled. When all three temperatures are known the total heat flux to the charge surface can be calculated using expression (6).

$$\dot{q}_{total} = \dot{q}_{rad\ gc}(\vartheta_g, \vartheta_c) + \dot{q}_{conv\ gc}(\vartheta_g, \vartheta_c) + \dot{q}_{rad\ wc}(\vartheta_w, \vartheta_c) - \dot{q}_{abs\ g}(\vartheta_g) \tag{6}$$

The calculation of the above mentioned heat fluxes is described in detail in [8].

Simulation experiment

The simulation experiment was implemented in industrial network system as shown in Figure 4a. The main process computer Dec α collects both the measuring data from the furnace through the PLC network TPL and the slab tracking parameters which are entered by the furnace operator. It is connected to Ethernet LAN. The real-time supervision system, which is also connected to LAN, transfers data at sampling intervals from the main process computer and then calculates the temperature field in the slabs. The calculated data are available through internet for real-time graphical representation. The supervision system operates under Debian Linux operating system running on a personal computer. It provides stable and fast environment for real–time operation of the supervision system.

The simulation parameters were: the sampling interval 60 s, the cross section of every slab was divided on $(M = 21) \times (N = 11)$ elements, the calculation interval was $\Delta t = 1\ s$, the average slab dimensions were: length 6 m, width 1 m, height 0.2 m. During the simulation many other parameters have to be known like the

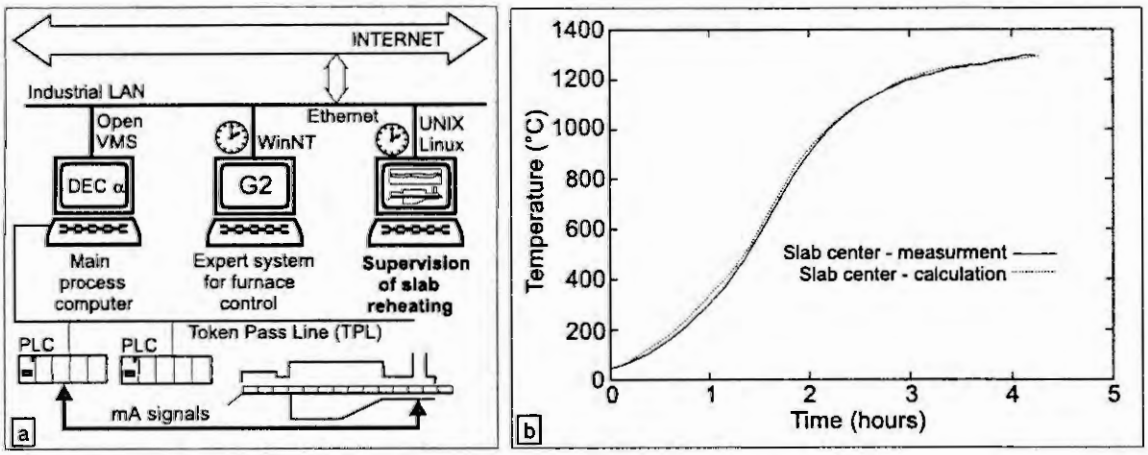


Figure 4: a) Implementation in industrial network system; b) Comparison between measurment and calculation

thermal properties of slab materials, the furnace dimensions, the thermal properties of furnace wall materials, which depend on each individual furnace and its charge.

The supervision system was verified by measurements on the real object. In the center of the slab the trail thermocouple was mounted. Figure 4b shows results of parallel simulation and test measurement during the slab reheating in the furnace after tuning of the model. The tuning of the model is necessary because of simplifying in the modelling phase. It is important that after the tunning all the values were inside the expected intervals for individual physical values.

Simulation results

The model gives many simulation results. Every reheated slab has its own reheating history file with temperature

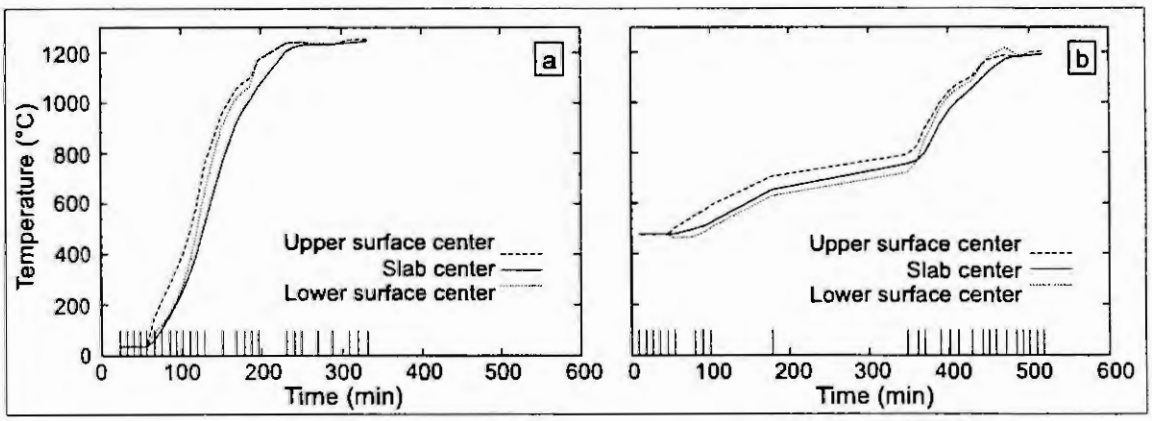


Figure 5: a)Short reheating of slab 49575; b)Long reheating of slab 50396

fields ($M \times N$) on every position during the reheating process in the furnace. Slab reheating process can be completely different for the same types of materials and dimensions. The comparison of reheating processes between the slab No.49575 and the slab No.50396 is shown on Figure 5. Slab No.49575 is cold at the beginning; its initial temperature is 20 °C and reheating process lasts 330 minutes, while slab No.50396 is at the beginning still hot (440 °C); its reheating lasts 520 minutes. Short vertical lines in the bottom of the diagrams represent times of pushes at which slabs change their position in the furnace. During the reheating of slab No.50396 there were two long delays which can be recognized by longer spaces between short vertical lines.

The model allows calculations of accumulated heat in each individual slab. Figure 6a shows accumulated heat diagram dependent on position in the furnace for the reheating processes of two slabs described earlier. There can be seen the difference in accumulation for cold and hot charged slabs. If the sum of accumulated heat in all slabs residing in the furnace in the sampling interval is divided by the chemical heat which enters the furnace by fuel, the instantaneous thermal efficiency of the furnace can be calculated. Figure 6b shows typical diagram of instantaneous thermal efficiency. There is strong correlation between pushing intervals and the efficiency. During very long delays, the efficiency can lower up to 0 %.

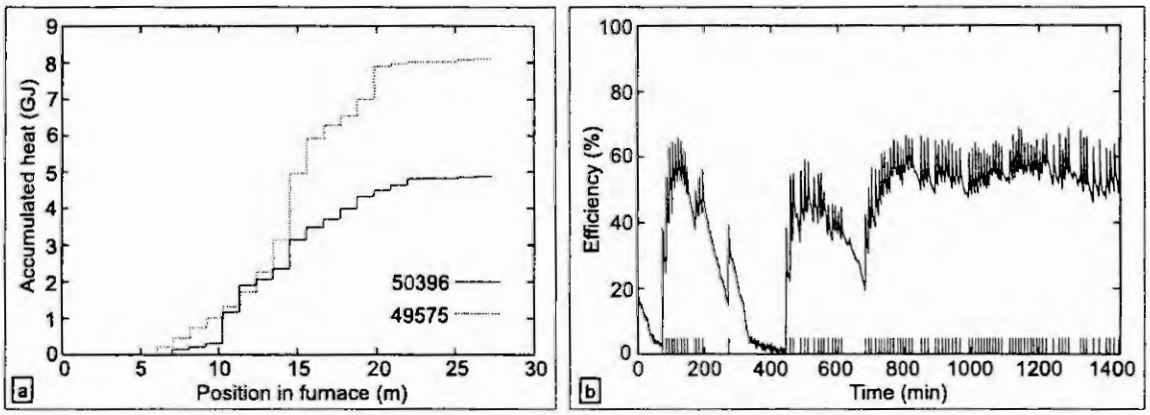


Figure 6: a)Accumulated heat depending on position in the furnace, slab No.50396 is hot charged, slab No.49575 is cold charged; b) Instantaneous thermal efficiency of the furnace

Summary

The described supervision system based on the mathematical model has been developed as a helping tool to the furnace operators. It allows monitoring of slab temperature fields in real-time during the reheating. It was implemented in an industrial network system. The informations produced by the monitoring system can be acquired on the Internet. For real-time calculation of heat exchange in the furnace, the Heiligenstædt simple three temperature model [3] is chosen. The heat transfer inside the slabs is calculated by two dimensional finite difference method. The model is stable even at long delays when sometimes charge temperature ϑ_c can exceed the inner furnace wall temperature ϑ_w. In the future the model is going to be accomplished by remote user friendly graphical interface for Windows platform.

References

1. Carpenter, D.G., Proctor, C.W., Temperature control and optimization of a reheat furnace using a distributed control system. Iron and Steel Engineer, August 1987, 44-49.
2. Glatt, R.D., and Macedo, f.X., Computer control of reheating furnaces. Iron and Steel International, December 1977, 381–396.
3. Heiligenstaedt, W., Wärmetechnische Rechnungen für Industrieöfen: Düsseldorf, Verlag Stahleisen m.b.H., 1966, 4 edn, 223–234.
4. Holander, F. and Huisman, R.L., On–line computer control for five zone reheating furnaces in a modern hot strip mill. In: International Conference on Iron and Steelmaking Automation II, Düsseldorf, April 1970, 143–152.
5. Holander, F. and Huisman, R.L., Computer Controlled Reheating Furnaces Optimize Hot Strip Mill Performance. Iron and Steel Engineer, September 1972, 43–56.
6. Hollander, F., Design, development and performance of on–line computer control in a 3–zone reheating furnace. Iron and Steel Engineer, January 1982, 44–52.
7. Hopkinson, C., Schofield, M., Adams, R., Monitoring and Control System in a Heavy Section Steel Mill. Metall, 53, Nr. 6, 1999, 320-322.
8. Kolenko, T., Glogovac, B. and Jaklič, A., An Analysis of a Heat Transfer Model for Situations Involving Gas And Surface Radiative Heat Transfer. Communications in Numerical Methods in Engineering, 15, 349-365, 1999.
9. Leden, Bo, A Control System for Fuel Optimization of Reheating Furnaces. Scandinavian Journal of Metallurgy, 15, 1986, 16–24.
10. Schurko, R.J., Weinstein, C., Hanne, M.K., Pellecchia D.J., Computer control of reheat furnaces: A comparison of strategies and applications. Iron and Steel Engineer, May 1987, 37–42.
11. Yoshitani, N., Ueyama, T. and Usui, M., Optimal Slab Heating Control with Temperature Trajectory Optimization In: IECON'94, 1994, 1567-1571.

INTEGRATED ENVIRONMENT FOR MODELLING, SIMULATION AND CONTROL DESIGN FOR ROBOTIC MANIPULATORS

Leon Žlajpah
Jožef Stefan Institute
Jamova 39, 1000 Ljubljana, Slovenia
leon.zlajpah@ijs.si

Abstract. In the paper an integrated environment for design of robotic controllers is described. It is based on the Planar Manipulators Toolbox for dynamic simulation of redundant planar manipulators. To be able to test the developed algorithms on a real robot, our system includes also a planar robot manipulator with four links and low gear actuators. The paper gives a short description of the Planar Manipulators Toolbox and the experimental system. The main advantage is the flexibility in fast prototyping of different algorithms in the field of control of robotic systems, especially for redundant manipulators. The tools are fully integrated in the MATLAB/SIMULINK and hence, a lot of standard tools are available for the analysis and control design. Using the real-time simulation it is possible to apply the developed controllers to a real robot without any additional coding.

1 Introduction

Simulation has been recognized as an important tool in designing new products, investigating its performances and also in designing applications of these products. Simulation allows us to study the structure, characteristics and the function of a system at different levels of details each posing different requirements for the simulation tools. As the complexity of the system under investigation increases the role of the simulation becomes more and more important. Advanced robotic systems are quite complex systems. Hence, the simulation tools can certainly enhance the design, development, and even the operation of robotic systems. Augmenting the simulation with visualization tools and interfaces, one can simulate the operation of the robotic systems in a very realistic way. Depending on the particular application different structural attributes and functional parameters have to be modelled. Therefore, a variety of simulation tools has been developed for the robotic systems which are used in mechanical design of robotic manipulators, design of control systems, off-line programming systems, to design and test of robot cells, etc. The majority of the robot simulation tools focus on the motion of the robotic manipulator in different environments.

A very important part of the robotic system is the control system. From the control viewpoint there are different control levels. The lowest level is the close-loop control and the next higher is the trajectory planning. The path planning and other more global control tasks are performed at higher levels. In the past, a lot of work has been done in the simulation of higher levels of control like path planning. For example, there exist different tools for planning of collision free paths, for teaching assembly operations [4], sensor-based operations [1]. Furthermore, almost all commercial robot systems are equipped with off-line programming systems [5]. For these purposes, the simulation of the kinematics and the structure of the robotic manipulator is important. However, when the lower levels of the control system are under investigation, also the dynamics of the robot manipulator becomes important.

For robotic systems, several simulation tools have been developed which can be used for control design: "A Robotic Toolbox" [2] and "A Toolbox for Simulation of Robotics Systems" [6] are implemented in MATLAB, and "Robotica" [3] is based on Mathematica. All these packages support general manipulator structures and as such they are rather complex and not appropriate for the real-time simulation. Furthermore, testing of different control algorithms on a real robotic systems is in general not very user friendly: the algorithms have to be rewritten for the real-time execution and the implementation details have to be considered. Therefore, we have developed a tool for analysis, design and testing of control algorithms for robotic manipulators especially for the redundant manipulators. The system consists of a simulation software and a real redundant robotic manipulator.

2 Structure of the integrated environment

In general, in the process of controller design different steps have to be performed. First of all, the system has to be modelled. In the next step, the control algorithm is developed. The first results are then obtained by the simulation. If the results are satisfactory, then in the final stage the control algorithms are tested on a real system. For this, a real-time code should be generated and implemented on the real system. The integration of all these steps, although essential, is very difficult. Namely, the different steps in the development of the controller require

the use of different methods for which different tools are needed. Hence, the results from one step to another have to be transferred often by hand. This bottleneck can be overcome if control design and testing is done in an integrated environment.

Our system consists of two major parts. The first one is a simulation package Planar Manipulators Toolbox for dynamic simulation of robotic manipulators and the other is an experimental planar robot manipulator with four links, revolute joints and low gear actuators. Fig. 1 shows the block diagram of the robotic system which has been developed.

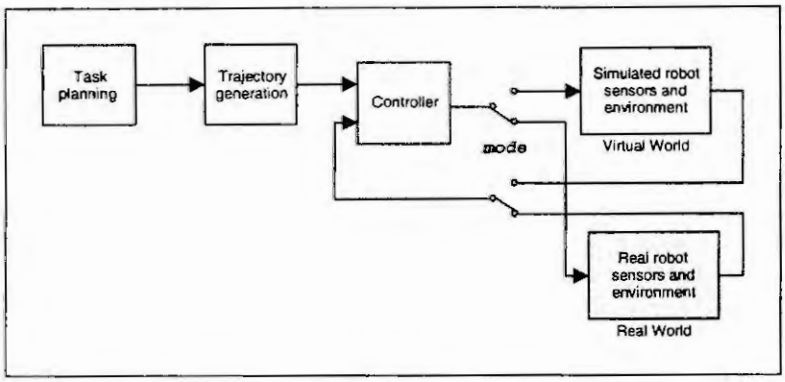


Figure 1: A functional block diagram of the integrated environment

As it can be seen from the figure, a very important feature of the environment is that it supports two modes of operation: "pure" simulation, where the robot, sensors and the environment are simulated , and "manipulator-in-the-loop" simulation, where a real robot with sensors working in the real world is included in the simulation loop.

The purpose of the "pure" simulation is to help the user in the design process to examine the behaviour of the system performing a particular task before it is carried out on a real system. In this way, the efficiency of the design is increased and the safety of the real robot is insured in greater extent. The other advantage of the simulation mode is that special situations can be examined which are hardly realized experimentally. Of course, the simulation has also some drawbacks. One of the drawbacks are errors due to the inaccurate modelling and the errors in the identification of model parameters. Although careful modelling can minimize these errors, a test on a real system is necessary for the final approval. This step is usually done separately using another control system. Hence, rewriting the developed algorithms into another language and/or for a different platform may be needed. For the implementation on the real system it is also necessary to consider different implementation details like interfaces, sampling frequencies, etc. As a consequence, the development time can be significantly prolonged, especially if the design process has to be repeated. A promising concept to overcome these problems and to integrate the design phase and the testing on a real system in one environment is the hardware-in-the-loop simulation. The main idea is to substitute one part of the simulation model by a real system during testing. We do this by using the "manipulator-in-the-loop" mode. Which part of the simulated system is substituted by a real one depends on what the user wants to test. Anyway, to have an efficient design environment the switching between simulation modes should be fast and without additional coding.

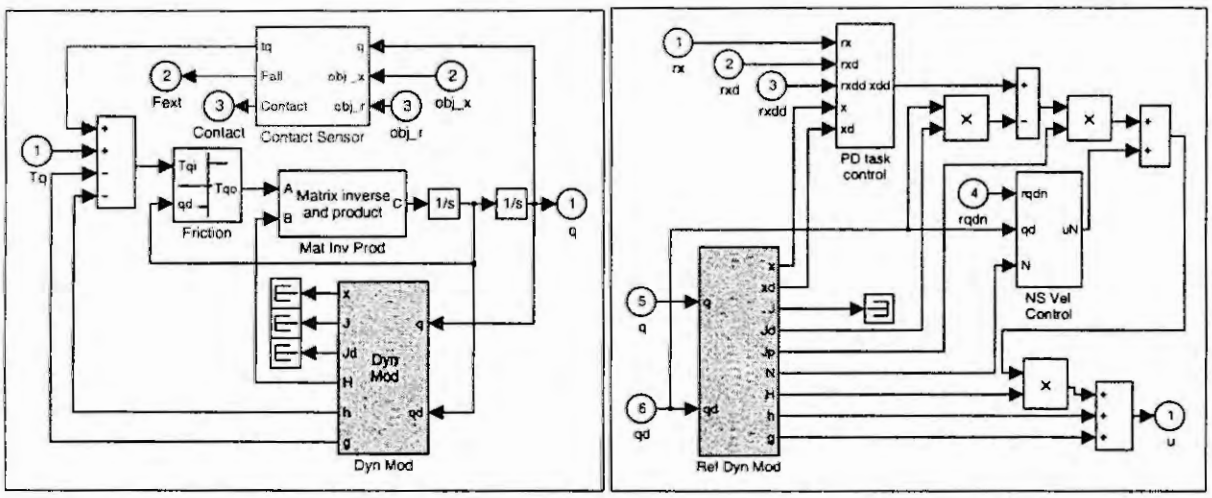
Prerequisites for the "hardware-in-the-loop" simulation is the ability of real-time simulation and additional hardware components which are needed for linking the real system and the simulation system together. In our case "hardware-in-the-loop" means "manipulator-in-the-loop", i.e. the model of the manipulator and its sensors are replaced by a real manipulator.

3 Planar Manipulators Toolbox

The Planar Manipulators Toolbox is a toolbox for simulation of n-R planar manipulators in MATLAB and SIMULINK [7]. MATLAB has been selected mainly due to its capabilities of solving problems with matrix formulations, easy extensibility, and because of the possibility to simulate in real- time, and to automatically generate real-time code. The toolbox is based on a kinematic and dynamic model of a planar manipulator with revolute joints. The detailed derivation of the manipulator models for this type of manipulators is given in [8]. As the complexity of the model for this particular type of manipulators increases slower with the number of DOF than in the case of general manipulators, the derived toolbox permits simulation of manipulators with many DOF within reasonable simulation times. The toolbox consists of several M-files for the calculation of the model and

other functions for planar manipulators which can not be created using the standard ones. The M-file functions are written in a straightforward manner to gain the understanding of the functions.

As an extension to MATLAB, SIMULINK adds many features for easier simulation of dynamic systems, e.q. graphical model building and selection of the integration method and parameters. To exploit additional features, we have developed several blocks and functions needed to create kinematic and dynamic models and to simulate the motion of n-R planar manipulators in SIMULINK. Combining these special blocks with other blocks subsystems representing kinematic or dynamic models are obtained. To enable the real-time simulation of manipulators with real manipulators in the loop using the Real-Time Workshop all S-functions have been rewritten into CMEX S-functions. In the Toolbox some common manipulator subsystems are prepared as simulation subsystem blocks. To gain the transparency the system is represented by the block structure with several hierarchical levels. Fig. 1 shows the typical robot system at the top level (from the control point of view) and Fig. 2 shows the dynamic model of a manipulator and a sensor detecting the object in the neighbourhood of the manipulator, and the task space and the null-space controller for a redundant manipulator.



(a) Model (b) Control

Figure 2: A block diagram representing (a) the dynamic model of a manipulator and a sensor detecting the object in the neighbourhood of the manipulator, (b) the task space and the null-space controller for a redundant manipulator

4 Experimental system

The main part of our experimental system is a laboratory manipulator (see Fig. 3) developed specially for testing of different control algorithms for robotic manipulators. The manipulator has four revolute DOF acting in a plane. As the task space is two-dimensional (x-y plane) the manipulator is redundant. Such configuration of the manipulator enables the testing of algorithms for redundant systems. We have selected the configuration with two redundant DOF because in our opinion one redundant DOF is not enough to test all important characteristics of redundant systems.

The link lengths are approximately 0.25m each. The manipulator is driven by AC motors and gears with low gear ratios (12 and 6). The servo drives have different control modes: the torque mode and the speed mode. Actually, the flexibility of the servo drives determines how the real manipulator can be included into the simulation system. For example, if a controller includes compensators for manipulator dynamics, then the control signals should represent joint torques and servo drives must be able to work in torque control mode. On the other hand, when control signals correspond to joint velocities servo drives must be in speed control mode. The motors are equipped with incremental encoders with 2500 pulses/rev. Note also that when using the hardware-in-the-loop additional gains should be included in the system. They aim is to obtain correct signal values for/from the interfaces and to make available in the simulation model the signals which can be measured on the real systems used. In this way the transparency of the system is very good.

The interface between the simulation systems and the manipulator consists of I/O boards with the corresponding software drivers at the control computer and servo drives at the manipulator side. For the control we use

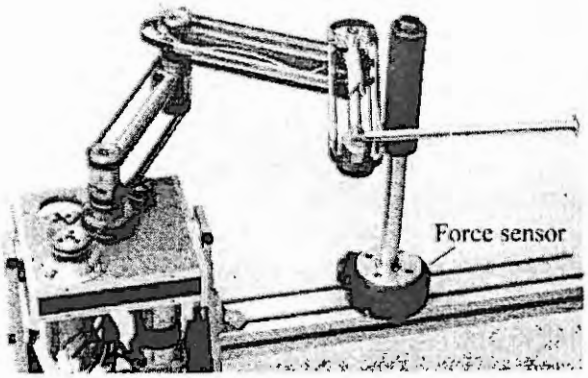


Figure 3: Experimental 4-R planar manipulator and a force sensor

a PC computer (Pentium II/200MHz). For linking the manipulator with the computer we have Lab-PC+ board (National Instruments), CYDDA06 board (CyberResearch) and for interfacing the incremental encoders our own PC-board based on LM628 (National Semiconductor). With this equipment sampling frequencies more than 1kHz can be achieved depending on the complexity of the control algorithms.

5 Conclusion

In the paper we present an integrated environment for controller design for a robot manipulator. The system consists of two subsystems. The first one is a simulation package Planar Manipulators Toolbox for dynamic simulation of n-degree of freedom planar manipulator based on MATLAB/SIMULINK and the other subsystem is an experimental planar robot manipulator with four links, revolute joints and low gear actuators. The paper gives a short description of the Planar Manipulators Toolbox and the experimental system. The emphasis is given to the connection of both subsystems into an integrated environment for the design and the testing. We show that the "manipulator-in-the-loop" simulation is a promising concept in the process of controller design for robotic manipulators as it enables the simulation of different control schemes and immediate testing on a real system. The Planar Manipulators Toolbox has proved tobe a very useful and effective tool for many purposes: kinematic simulation, dynamic simulation, analysis and synthesis of control systems, trajectory generation, etc. It is very easy to extend and to adapt the simulation package to different requirements. Thus, it should be of interest to the researchers involved in the development of advanced robot control systems.

6 References

[1] C.X. Chen, M.M. Trivedi, and C.R. Bidlack. Simulation and graphical interface for programming and vizualization of sensor-based robot operation. In *Proc. Intl. Conf. On Robotics and Automation*, pages 1095 –1101, Nice, France, 1992.

[2] P. I. Corke. A Robotics Toolbox for MATLAB. *IEEE Robotics & Automation Magazine*, 3(1):24 – 32, 1996.

[3] J.F. Nethery and M.W. Spong. Robotica: a Mathematica package for robot analysis. *IEEE Robotics & Automation Magazine*, 1(1), 1994.

[4] H. Ogata and T. Takahashi. Robotic assembly operation teaching in a virtual environment. *IEEE Trans. on Robotics and Automation*, 10(3):391 – 399, 1994.

[5] G. Schneider and A. Kazi. Trends in applied robot controller development. In *Proceedings of 8th Int. Workshop on Robotics in Alpe–Adria–Danube Region*, pages 11 – 16, München, 1998.

[6] D. Surdilovic, E. Lizama, and J. Kirchhof. A Toolbox for Simulation of Robotic Systems. In F. Breitenecker and I. Husinsky, editors, *EUROSIM '95 Simulation Congress*, pages 693 – 698, Vienna, 1995. Elsevier Science.

[7] L. Žlajpah. Planar Manipulators Toolbox: User's Guide. Technical Report DP - 7791, Jožef Stefan Institute, 1997. URL: `http://www2.ijs.si/~L./planman.html`.

[8] L. Žlajpah. Simulation of *n*-r planar manipulators. *Simulation Practice and Theory*, 6(3):305 – 321., 1998.

MULTIVARIABLE PLANT MODELLING BY COMBINATION OF DIFFERENT APPROACHES

Maja Atanasijević-Kunc, Gregor Klančar, Srečko Milanič, Rihard Karba, Borut Zupančič
Faculty for Electrical Engineering, Tržaška 25, 1000 Ljubljana, Slovenia

Abstract. In the paper we investigate and compare the efficacy of different modelling approaches where special attention is devoted to hybrid modelling possibilities for real multivariable plant. It is well known that for control design usually good models are needed which is especially true when we are dealing with multivariable systems and demanding design goals which can involve also fault detection aspects. So we can expect that the reachable or achieved control quality is in the great measure dependent on the quality and the reliability of the model. Such situations can also involve considerable changes in system operation regarding open-loop and different closed-loop configurations. Due to this we are presenting and evaluating step by step model improvement starting from open-loop operation and theoretical modelling, while in further steps different controller structures are used for improvement and verification of modelling results. Obtained results are analysed and compared with measurement data.

Introduction.

Modelling procedure can be regarded as a complex one as usually in cyclic procedure a set of models is obtained for description of system behaviour. Each model can of course be informative in certain design step. Design steps are usually closely related with the purpose of model usage. When building the model input and output signals are usually compared with measurement data and on the basis of these results evaluation price is obtained. This is quite often the case also when the model is to be used for closed - loop design. But when the matching of model with system behaviour is not good, this difference can in the close - loop be hidden regarding input - output relations due to the controller action. Even quite different controllers can from input - output point of view produce similar signals. The most important difference can in this case be observed through control signals. In general we can expect that more demanding controllers would cause greater difference between the system and model control signals. This difference can be important from different points of view. It can be used for (step by step) model improvement. If mismatching between the model and the system is smaller the quality of reachable controller action can be improved as the needed robustness limits can be smaller. If the matching of all signals is better also fault detection can be improved as different algorithms can in great measure be dependent also on this information.

We are presenting the described ideas for the system of three coupled tanks (made by AMIRA [1]) as they are shown in Fig. 1.

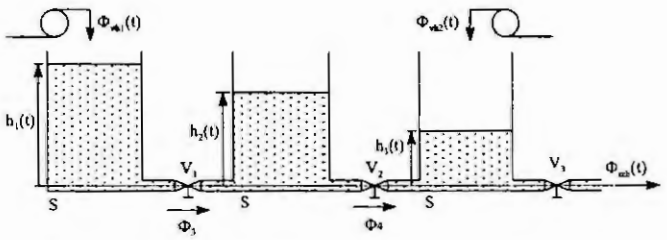


Figure 1: Schematic representation of the process

In the first step nonlinear model (NM) was built through theoretical approach where some frequently used assumptions have been introduced, like: the water flows through the valves are square root functions of pressure difference and activators and sensors have linear characteristics in the whole operating range. The obtained nonlinear description was tested in open and closed - loop, where first only the controller which consists of two SISO - PI was used:

$$\underline{\underline{G}}_{PI}(s) = \begin{bmatrix} -3.6230 + \frac{-0.0181}{s} & 0. \\ 0. & -10.9200 + \frac{-0.1092}{s} \end{bmatrix} \quad (1)$$

Further modelling steps

Additional modelling efforts were realised in the following directions:
1. Recurrent (two-layer perceptron) artificial neural network (ANN) was tuned (time delay units are used to learn a system's dynamic), which in neural network terminology means that future network inputs will depend on present and past network outputs. First only open-loop signals were included into learning set and in the next step the model was evaluated using closed-loop control signals. We have to point out that learning open loop signals were simply the set of step signals as are frequently used for theoretical approaches. The system was driven first to the operating point, then the properties around this point were investigated and finally the shut down was realised. As such signals can be problematic from identification aspects (but more demanding signals can be problematic in process industry), in the second step also closed-loop signals were added in learning set and the behaviour of the network was further improved.
2. For the improved model more demanding controller was designed in the following form [2]:

$$\begin{aligned} \underline{u} &= \underline{\underline{F}}\,\underline{x} + \underline{\underline{G}}\,\underline{\omega} \\ \underline{\omega} &= \begin{bmatrix} 1+\frac{0.1}{s} & 0. \\ 0. & 1+\frac{0.1}{s} \end{bmatrix} (\underline{r}-\underline{y}) \\ \underline{\underline{F}} &= \begin{bmatrix} 0.8242 & 1.3846 & 0 \\ 0 & 1.3043 & -0.1304 \end{bmatrix}, \quad \underline{\underline{G}} = \begin{bmatrix} -2.1978 & 0 \\ 0 & -2.1739 \end{bmatrix} \end{aligned} \tag{2}$$

Further evaluation and improvement steps were realised with this controller (considerably faster closed-loop system) and very demanding reference signals.

In the next steps we have investigated the possibilities of different neural networks to improve nonlinear analytical and obtained ANN model properties. In these steps the learning signals were realised on the basis of the difference between the existing model and measurement signals, again for open and closed-loop realisations. Here additional difficulty arises as these differences are relatively small. But such situations are of practical importance also in the cases where through longer operating time the system can slightly change and the model accuracy is not satisfactory any more. To avoid the modelling from the very beginning the existing model can be improved in such a manner that additional structure is added to the existing one. In this paper the following situations are inspected.

3. Non-recurrent ANN (NANN) structure was added. The regressors in this step consist of inputs and outputs of existing model (nonlinear model or ANN model from the second step), where the number of delayed units must be at least as large as is the order of the process. The structure is multi-layer perceptron (MLP) with ten hyperbolic tangent hidden neurons (in average depending on combination) and a linear output neuron. Each of the three MISO structures was trained with Levenberg-Marquardt algorithm [3] and realised in the parallel with the existing one.

4. Recurrent ANN (RANN) structure was added in parallel so the network has feedback through the choice of regressors meaning that to regressors from the third step the delayed network predictions are added. Recurrent MLP structure makes neural model much more powerful so for all three MISO models around five hidden neurons were sufficient.

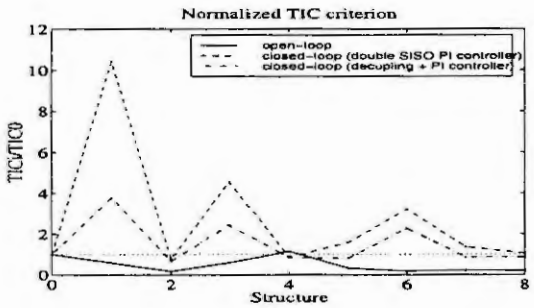
5. Gaussian Radial Basis (GRB) network structure was added in parallel where hidden layer consists of locally-tuned activation functions. It is constructed by first using k-means clustering and determination the widths of GRB units. The output linear layer weights are obtained with least square calculation. Around forty hidden neurons were enough to capture behaviour of the existing model uncertainties. This type of ANN has like perceptron good interpolation abilities but its extrapolation is much worse beyond its region of training data [4]. This makes them less useful when the existing model uncertainties are noticeable over the whole operating range. The reason lies in the local hidden units and learning algorithm.

6. Non-recurrent ANN structure was added in serial (SANN) to existing model. The MLP was trained to perform the transformation between process and existing model outputs, where the only condition was

equality of process and existing model inputs. This condition is more or less satisfied in a close-loop with controller action. The network architecture was MIMO with four hidden and two output neurons.

Summary

All the mentioned structures and additional structures were tested in combination with analytical nonlinear and ANN model in open and closed-loop and compared with the measurement data. Evaluation results are presented in Fig. 2. For each structure Theil's inequality coefficients (TIC) were first defined due to the measurement data and then they were normalised on the fitting result of nonlinear model. This means that the values which are smaller then 1 are better then the fitting of nonlinear model while the others are worse.



No.0: nonlinear model(NM), **No.1:** NM+NANN, **No.2:** NM+RANN, **No.3:** NM+GRB, **No.4:** NM+SANN, **No.5:** ANN, **No.6:** ANN+NANN, **No.7:** ANN+RANN, **No.8:** ANN+GRB

Figure 2: Evaluation results

On the bases of the obtained results and analysis we can conclude the following. Non-recurrent ANN structure can in parallel realisation be successful in open-loop operation (fault detection) but is unsuitable for the closed-loop operation as it can't cover the needed dynamics of additional outputs. Such situation is illustrated in Fig. 3 for the combination with ANN in closed-loop with the more demanding controller as is described with (2). Simulation results of nonlinear model and measurement data are compared with this structure. In the upper two figures the control signals are presented, while in the three lower figures we observe the level changes in all the tanks. We can see that the reference signals are very demanding as control signals are covering the whole operation range. Better behaviour can be expected if it is used in serial with the existing model. GRB network structure has advantages in parallel combination to existing model, when its uncertainties are not noticeable over the whole operating range but only locally. Recurrent ANN has great potential in open and closed-loop behaviour, as self-standing or additional model. As additional structure it was tested in combination with previously tuned ANN and nonlinear model. The last situation is illustrated in Fig. 4 for the combination with nonlinear model in closed-loop with the more demanding controller. This hybrid structure was also very successful in step by step tuning where further combinations with control design and fault detection are of great interest. It seems that such approach can be very effective in all situations where the lack of measurement possibilities disables theoretical definition of all system properties.

References

1. AMIRA, DTS 200 Laboratory Setup Three - Tanks - System, Manual, 1995.
2. M. Atanasijević-Kunc, R. Karba, B. Zupančič, A. Belič: The use of genetic algorithms in modelling the multivariable control problem, Proceedings of Eurosim'98 Simulation Congress, Vol. 3, pp. 548-552.
3. M. Norgaard: Neural Network Based System Identification Toolbox, Technical University of Denmark, Lyngby, 1997.
4. M. Thompson, M. Kramer: Modeling Chemical Processes Using Prior Knowledge and Neural Networks, AIChE Journal, Vol. 40, No. 8, pp. 1328-1340, August 1994.

5. S. Milanič, M. Atanasijević-Kunc, R. Karba, B. Zupančič: Combination of two approaches to modelling of pressure - level pilot plant, Proceedings of IMACS Symposium on Mathematical Modelling, Vienna, Austria, pp. 63-68,1997.
6. S. Milanič, M. Atanasijević-Kunc, R. Karba, B. Zupančič: From theoretical modelling to a Hybrid neural model of pilot plant, Proceedings of the 15^{th} IMACS World Congress, pp. 209-214, Berlin 1997.
7. Neural Network Toolbox User's Guide, MathWorks, Natick, 1997.

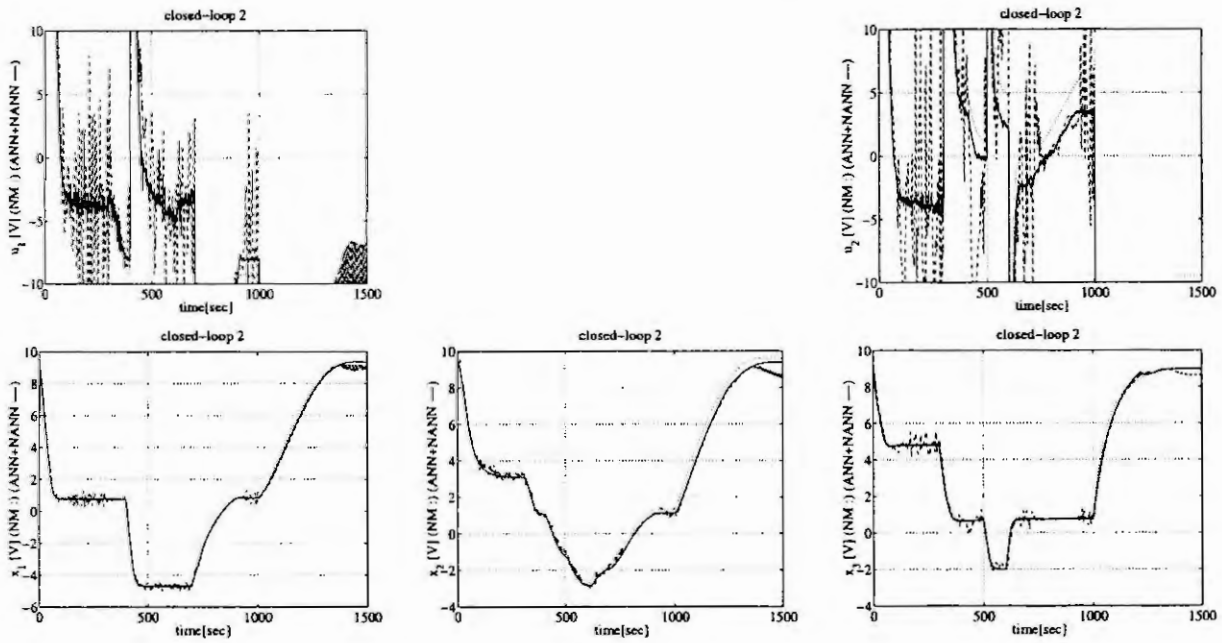


Figure 3: Close-loop responses of the process (-), nonlinear model (..) and recurrent ANN in combination with NANN structure in parallel (- -)

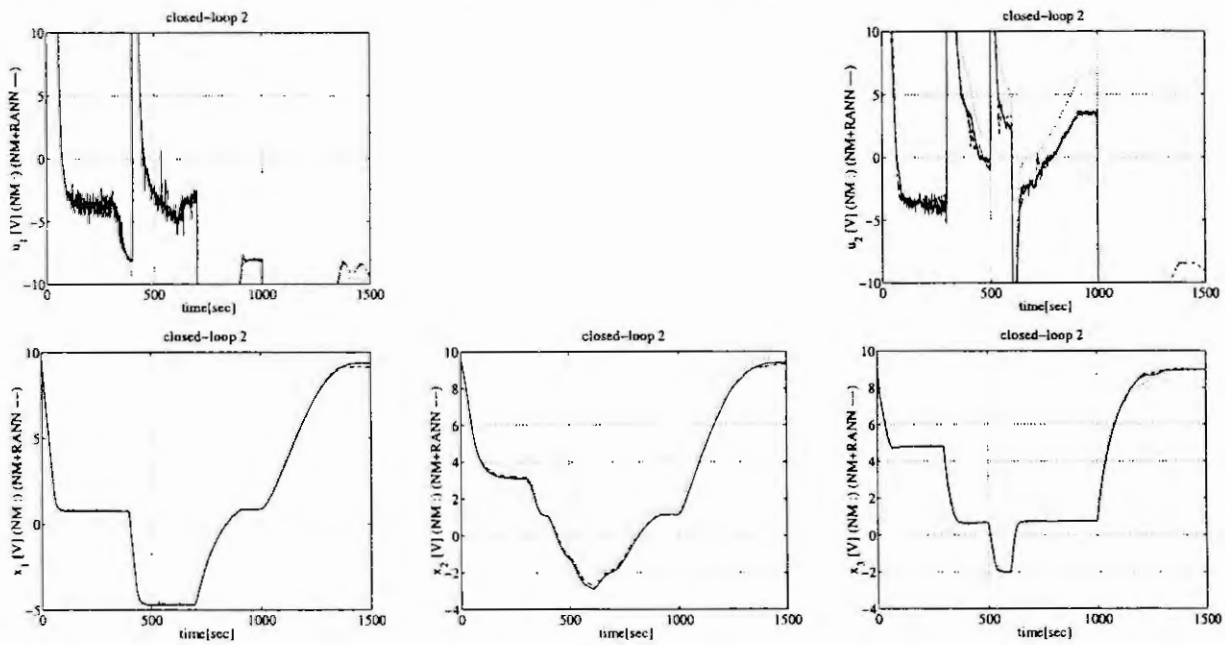


Figure 4: Close-loop responses of the process (-), nonlinear model (..) and nonlinear model in combination with RANN structure in parallel (- -)

Fuzzy modeling and model based control with use of a priori knowledge

J. Abonyi[1,2], R. Babuška[1], L.F.A. Wessels[1], H.B. Verbruggen[1], F. Szeifert[2]
[1]Delft University of Technology, Department of Information Technology and Systems
Control Laboratory, P.O.Box 5031 2600 GA Delft, The Netherlands
[2]University of Veszprem Department of Chemical Engineering Cybernetics
P.O. Box 158, H-8201, Hungary

Abstract. In order to solve the problem of model based control arising from the process model has to be obtained by using small amount and different type of available information, a fuzzy modeling framework has been developed for the utilization of *a priori* knowledge. The proposed modeling approach transforms the different types of information into the structure of the model (fuzzy rule base), constraints defined on the parameters and variables, dynamic local model or data, and steady-state data or model. This modeling step is followed by an optimization procedure based on these transformed information. The paper describes one element of this framework that was developed to use prior knowledge in constrained adaptation of the rule consequences of Takagi-Sugeno fuzzy models. Experimental results have been obtained for a laboratory setup consisting of two cascaded tanks. It has been shown that by using constrained adaptation, good control performance can be achieved for a nonlinear, time-varying process.

1 Introduction

A critical step in the application of model-based control algorithms is the development of a suitable control relevant model. These difficulties stem from a lack of information about the process to be controlled due to the complexity of the system. In order to solve this problem, there is a tendency to blend information of different nature: experience of operators and designers, measurements and first principle knowledge formulated by mathematical equations. The aim of this paper is to present a fuzzy modeling framework that is suitable for the use of such information to generate control-relevant process models.

The paper is organized as follows. In Section 2 the developed fuzzy modeling framework is reviewed. Section 3 describes one element of this framework developed to use prior knowledge in constrained adaptation of the rule consequences of Takagi-Sugeno (TS) fuzzy models [6]. An application of the proposed method to real-time predictive control of liquid level in a tank is also presented in this section. Conclusions are given in Section 4.

2 Using *a priori* knowledge in control relevant fuzzy modeling

Prior knowledge can be used both implicitly and explicitly in identification for control. During a general identification procedure, the prior knowledge is used implicitly in the design of the excitation signal and the determination of the model structure.

The explicit use of *a priori* knowledge means the direct integration of the knowledge into the model. The relevant a priori information can be obtained from different sources: mechanistic knowledge obtained from first-principles (physics and chemistry), empirical expert knowledge expressed by linguistic rules, and measurement data obtained during normal operation or identification experiment.

In general, different modeling paradigms should be used for an efficient utilization of these different sources of information. This means, if we have good mechanistic knowledge about the process, this can be transformed into white box model described by analytical (differential) equations. If we have information like human experience described by linguistic rules and variables, the application of rule-based approaches like fuzzy logic is more appropriate. There may be situations, where the most valuable information comes from input-output data. In this case, the application of black box models is the best choice. These black box models are especially valuable, when an accurate model of the process dynamics is needed.

Unfortunately, the real situation is clearly not any of the previously mentioned approaches, but rather a combination of them. For instance, fuzzy identification is an effective tool for the approximation of uncertain nonlinear

systems on the basis of measured data [4]. However, data-driven identification techniques alone sometimes yield unrealistic models in terms of steady-state characteristics, local linear behavior, or physically impossible parameter values. Typically, this is due to the insufficient information content of the identification data set, errors in the data, and experiments of limited duration. This example shows, the modeller has only small amount and different type of information to build the model. Therefore, in order to be able to employ as much knowledge as possible, there is a need for a modeling environment where different modeling paradigms can be used simultaneously to attack different sides of the problem. Hence, incorporation of a priori knowledge into the fuzzy identification for control is a challenging and important task, which motivated the development of the proposed modeling framework depicted in Figure 1.

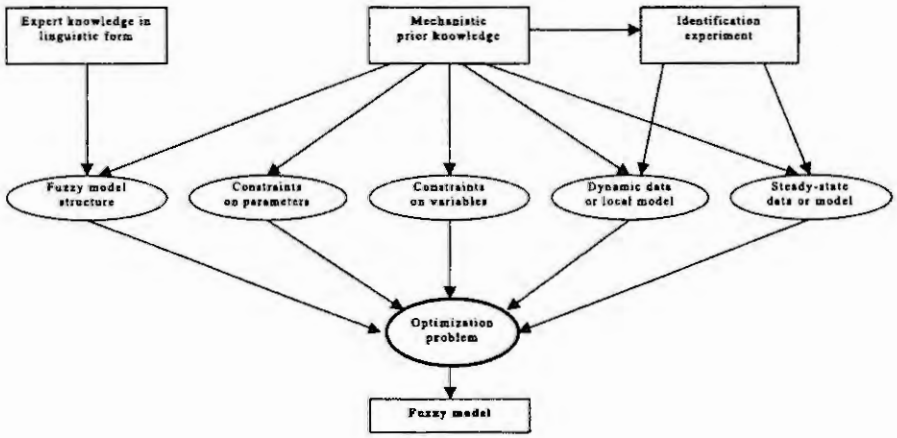


Figure 1. *Framework for a priori knowledge based control relevant fuzzy modeling.*

As Figure 1 shows, the proposed modeling approach transforms the available information into the structure of the model (fuzzy rule base), constraints defined on the parameters and the variables, dynamic local model or data, and steady-state data or model. This modeling step is followed by an optimization procedure what is based on these transformed information.

As a combination of a fuzzy model that represents the stationer behavior and *a priori* knowledge based gain independent impulse response of the system Hybrid Fuzzy Convolution Model [3] and Wiener Convolution Model [2] have been developed. This block-oriented approach is useful, if the modeller has *a priori* knowledge about the steady-state or the gain-independent dynamic behaviour of the process.

For the incorporation of prior knowledge into data-driven identification of dynamic fuzzy models of the Takagi-Sugeno type, a constrained identification algorithm has been developed [1]. This approach is based on the transformation of the *a priori* knowledge about stability, bounds on the stationary gains, and the settling time of the process into linear inequalities on the parameter set of the fuzzy model, similar to [8, 7]. This constrained identification is useful, because the TS fuzzy model is often overparametrized, hence explicit regularization, like penalties on non-smooth behaviour of the model and application of *a priori* knowledge based parameter constraints can dramatically improve the robustness of the identification algorithm, eventually leading to more accurate parameter estimates [5].

As the next section will show, this constrained identification algorithm can be successfully applied in the on-line adaptation of fuzzy models.

3 Example: adaptive model predictive control of a laboratory liquid level process

The dynamic Nonlinear AutoRegressive with eXogenous input (NARX) model is frequently used within many nonlinear identification methods, such as neural network models and fuzzy models. The NARX type TS fuzzy model interpolates between local linear, time-invariant (LTI) ARX models as follows:

$$R_j: \textbf{ If } z_1(k) \text{ is } A_{1,j} \textbf{ and } \ldots \textbf{ and } z_n(k) \text{ is } A_{n,j} \textbf{ then } y^j(k+1) = \sum_{i=1}^{n_y} a_i^j y(k-i) + \sum_{i=1}^{n_u} b_i^j u(k-i-n_d) + c^j .$$

where j denotes the rule index, $z(k)$ is a scheduling vector that usually a subset of the previous outputs $y(k-i)$ and inputs $u(k-i-n_d)$, and $A_{i,j}$ is the fuzzy set in jth rule for the ith input.

The output of the Takagi-Sugeno fuzzy model is linear in its consequent parameters. Hence, the model can be formulated as:

$$y(k+1) = \sum_{j=1}^{S} \phi^T(k)\beta_j\theta_j + e(k), \tag{1}$$

where S is the number of the rules, β_j is the validity of the local model, $\phi(k)$ is the regressor vector, $\phi(k) = [y(k),\ldots,y(k-n_y),u(k-n_d),\ldots,u(k-n_u-n_d),1]$, θ_j is the parameter vector of the j-th local model (rule), $\theta_j = \left[a_1^j,\ldots,a_{n_y}^j,b_1^j,\ldots,b_{n_u}^j,c^j\right]$, and $e(k)$ is a zero-mean white noise sequence.

The unconstrained weighted recursive least squares (RLS) estimate of the parameters is:

$$\theta_j(k) = \theta_j(k-1) + \frac{P_j(k-2)\phi(k-1)\beta_j\left[y(k)-\phi^T(k-1)\theta_j\right]}{\alpha_j(k-1)+\beta_j\phi^T(k-1)P_j(k-2)\phi(k-1)} \tag{2}$$

$$P_j(k-1) = \frac{1}{\alpha_j(k-1)}\left[P_j(k-2) - \frac{\beta_j P_j(k-2)\phi(k-1)\phi^T(k-1)P_j(k-2)}{\alpha_j(k-1)+\beta_j\phi^T(k-1)P_j(k-2)\phi(k-1)}\right] \tag{3}$$

where P_j is a matrix proportional to the covariance matrix, and α_j is a scalar forgetting factor of the j-th rule adaptation. For time varying systems, in order to give greater weighing to more recent data, RLS with exponential data weighting ($0.8 < \alpha_j(k) \leq 1$) is used.

As it was mentioned in the previous section, the proposed framework transforms the *a priori* knowledge is into inequality constraints, formulated as:

$$L_j\theta_j \leq c_j\,. \tag{4}$$

The constrained solution of the recursive least squares identification can be obtained by the optimal projection of the unconstrained solution [7]:

$$\theta_j^c = \theta_j - P_j L_j^T \lambda_j, \tag{5}$$

where θ_j denotes the unconstrained RLS solution (2), while θ_j^c denotes the constrained solution. The λ_j vector of Lagrange multipliers associated with the equality and inequality constraints [7].

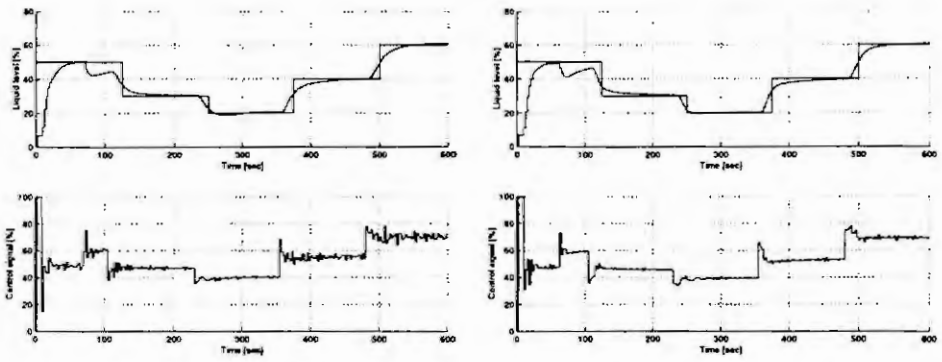
This approach has been applied in real-time model predictive control of liquid level in a two-tank system. The laboratory process, consists of two cascaded tanks. The description of the setup is given in [1].

As model-based controller, a model predictive controller (MPC) was employed to control the process. The prediction and the control horizons of the MPC controller are selected to be $H_{p1} = 3$, $H_{p2} = 8$, and $H_c = 2$. The move suppression coefficient is $\lambda_w = 0.05$. The constraints on the control signal are set to $\Delta u_{\max} = -\Delta u_{\min} = 0.25$, $u_{\max} = 1$, $u_{\min} = 0$.

The fuzzy model has the following structure. The input variables of the model were selected on the basis of prior knowledge. Since the process can be modeled approximately as a second-order system with a time delay, the regressor of the model is chosen to be $\phi(k) = [y(k), y(k-1), u(k-2), 1]$. Because the source of the nonlinearity is the level-dependent outflow from the tank, the antecedent variable of the fuzzy model is chosen to be $z(k) = y(k)$. Five fuzzy sets were defined on the antecedent universe, $y(k)$. This means, the fuzzy model consists of five local linear models. Based on the range of the liquid level and after some manual tuning, the cores of the fuzzy sets were selected to be $\{0.06\ 0.2\ 0.35\ 0.5\ 0.82\}$.

The fuzzy model was adapted in two ways. (1) No prior knowledge was involved during the adaptation, i.e., standard recursive least squares technique was used. (2) Prior knowledge on process stability, minimal and maximal gain and the settling time of the local model were assumed, and the proposed constrained adaptation method was used.

In order to illustrate the advantages of the proposed algorithm, the following experiment was designed. At the 75th second of operation an unmeasured process disturbance introduced by opening a bypass valve. Standard and constrained RLS were applied for the adaptation of the fuzzy model. Because, the constrained adaptive algorithm forces the parameters of the local models to be in the desired ranges given by the prior knowledge, by using *a priori* knowledge in the adaptation, the achieved control performance improves more than with the standard RLS used (Figure 2). Hence, the resulted control signal is more smooth (Figure 2(b)).



(a) Free adaptation *(b) Constrained adaptation*

Figure 2. *Performance of the adaptive predictive controller*

4 Conclusions

A critical step in the application of model-based control algorithms is the development of a suitable model of the process dynamics. These difficulties stem from a lack of knowledge or information about the process to be controlled due to the complexity of the system. In order to solve this problem a new modeling framework has been introduced for control relevant fuzzy modeling. The approach transforms the different type of information into the structure of the model (fuzzy rule base), constraints defined on the parameters and the variables, dynamic local model or data, and steady-state data or model. This modeling step is followed by an optimization procedure what is based on these transformed information.

As an example of this approach, a model-based adaptive control algorithm has been presented. The advantage of the algorithm is that by constraining the parameters of the local linear models, it is possible to speed up the adaptation and avoid unrealistic model parameters that could result in a poor control performance. Experimental results have been obtained for a laboratory setup consisting of two cascaded tanks. It has been shown that by using constrained adaptation, good control performance can be achieved for a nonlinear, time-varying process.

References

[1] J. Abonyi, R. Babuska, M. Setnes, H.B. Verbruggen, and F. Szeifert. Constrained parameter estimation in fuzzy modeling. In *Proceedings of FUZZ-IEEE'99*, pages 951–956, Seoul, Korea, August 1999.

[2] J. Abonyi, A. Bodizs, L. Nagy, and F. Szeifert. Predictor corrector controller using wiener fuzzy convolution model. *Hungarian Journal of Industrial Chemistry*, 27(3):227–233, 1999.

[3] J. Abonyi, L. Nagy, and F. Szeifert. Hybrid fuzzy convolution modelling and identification of chemical process systems. *International Journal of Systems Science*, to appear, 1999.

[4] R. Babuška. *Fuzzy Modeling for Control.* Kluwer Academic Publishers, Boston, 1998.

[5] T.A. Johansen. Identification of non-linear systems using empirical data and a priori knowledge – an optimisation approach. *Automatica*, 32:337–356, 1996.

[6] T. Takagi and M. Sugeno. Fuzzy identification of systems and its application to modeling and control. *IEEE Trans. Systems, Man and Cybernetics*, 15(1):116–132, 1985.

[7] W.D. Timmons, H.J. Chizeck, and P.G. Katona. Parameter-constrained adaptive control. *Ind. Eng. Chem. Res.*, 36:4894–4905, 1997.

[8] H.J.A.F Tulleken. Gray-box modeling and identification using pysical knowledge and Bayesian techniques. *Automatica*, 29:285–308, 1993.

MODELLING AND CONTROLLER DESIGN

D P Atherton and S Majhi
School of Engineering, University of Sussex
Falmer, Brighton BN1 9QT UK *d.p.atherton@sussex.ac.uk*

Abstract. The paper addresses the problem of the information required to tune the parameters of a fixed controller, typically PID, for a single loop system. Examples show the value of using the plant critical point. Its estimation by a relay autotuning test followed by the use of either the approximate describing function or exact limit cycle analysis is discussed. Situations where use of the describing function analysis may give poor results are also mentioned.

1. Introduction.

Analysis and design of control systems is typically taught to students assuming that a linear mathematical model, usually a transfer function initially, is available for the plant. This is, of course, a major assumption since in many practical situations it is not easy to find a good mathematical model and further if one could it would almost certainly be nonlinear. The linear approach, despite its deficiencies, is adopted because one can develop sound mathematical theories for the controller design so that the closed loop system behaviour meets certain specifications.

Given therefore that meeting given specifications is the objective of control system design the question which needs to be answered is what do we need to know about the plant in order to achieve them? What therefore is good information about the plant which can be determined easily so that a controller can be designed to achieve a satisfactory result?

Another major difficulty in design is translating the desired system properties, which may refer to process efficiency or emissions, into performance criteria applicable to control system design. The criteria which typically result usually relate to steady state performance and the response to input set point or disturbance changes. Interestingly two of the most widely used linear design approaches, namely open loop frequency response shaping and pole placement, only address these problems indirectly. Neither, for instance, change if the controller is moved from the forward path to the feedback path yet, since the closed loop zeros are different, the closed loop step response changes. Thus when using these design approaches iterative techniques have to be used to meet precise performance on step response criteria.

In the next section the relay autotuning approach is outlined and the merits of using limit cycle analysis based on the describing function and exact approaches discussed. Section 5 discusses controller tuning based on critical point information and is followed by an example and some conclusions.

2. Autotuning.

The advent of microprocessor based controllers has meant that the controller can contain not only a relatively complex control algorithm but that it can also perform additional tasks. These include features such as procedures for fault diagnosis or controlling an experiment which will enable it to obtain good controller parameters. This latter task may be referred to as autotuning and a simple procedure is relay autotuning. The approach may be regarded as an automated Ziegler-Nichols (Z-N) test with the proportional mode of the controller replaced by a relay. This test, which for most process control type plants will produce a limit cycle in the loop, has advantages and disadvantages compared with the Z-N approach. The major advantages are the relay ouput levels control the limit cycle amplitude and there is no trial and error to obtain the value of the proportional gain. The disadvantage is that the simplest procedure to get information is typically to use the approximate describing function (DF) method for the analysis. If the relay is ideal with output levels $\pm h$ then DF analysis yields the equation

$$1 + (4h/a\pi)G(j\omega) = 0 \qquad (1)$$

where $4h/a\pi$ is the DF of the relay and $G(j\omega)$ the plant frequency response. Since the DF has no phase shift this equation yields a frequency ω_c, where $\angle G(j\omega_c) = -180°$, for the limit cycle frequency, and the critical gain, K_c, of the plant, namely the gain margin expressed as a linear gain rather than dB, is $4h/a\pi$. Strictly speaking based on DF theory 'a' is the amplitude of the fundamental frequency of the limit cycle but for ease of measurement is usually taken to be half the peak to peak limit cycle amplitude, A. Thus from this simple test and the measurement of two limit cycle parameters, namely the amplitude and frequency, one can estimate ω_c and K_c. This result raises two questions, namely how accurate are the estimates for K_c and ω_c and how well can one design a controller based on these two parameters?

Since the accuracy of the describing function approach depends on how near the limit cycle is to being sinusoidal it is worth considering a first order plus dead time (FOPDT) process with transfer function $Ke^{-sTd}/(1+sT)$, since this is a poor filter and the actual limit cycle consists of two exponentials with a discontinuous derivative at the changeover. From the transfer function the critical frequency is given by

$$\omega_c T_d + \tan^{-1}\omega_c T = 180° \tag{2}$$

which can be written

$$\rho = (180° - \tan^{-1}\omega_c T)/\omega_c T \tag{3}$$

where $\rho = T_d/T$. An exact analysis for the limit cycle frequency, which is discussed later, gives

$$\rho = \log [1/(1-\tanh(\pi/2\omega_0 T)] \tag{4}$$

The difference between ω_0 and ω_c is always positive and as ρ varies has a maximum value of approximately 4.5% when $\rho \sim 2.5$. For ρ large the limit cycle is approximately a square wave and for ρ small approximately triangular, so measurement of the amplitude A can overestimate or underestimate the fundamental component [1]. Use of this information can be useful in correcting the measured values of ω_0 and A to give better values for use in the describing function expressions. Since, for example, when ρ tends to zero the limit cycle is near triangular, which is the situation for a pure integrating plant Ke^{-sTd}/s, if a is taken as $8A/\pi^2$, which is the fundamental of a triangular waveform, for use in eqn (1), exact results will be obtained for K_c and ω_c. The latter situation results because all the harmonics in a triangular waveform are in phase, so that the fundamental does not undergo a phase shift through the relay, and $\omega_c = \omega_0$. For a second order plus dead time process (SOPDT) because of the additional filtering more accurate results are given by the DF method for K_c and ω_c. It can probably be argued, bearing in mind that more sophisticated measurements, time or data processing would be required for other procedures giving more accurate information, the autotuning approach is justifiable for practical applications to processes with transfer functions similar to the FOPDT.

There are, however, situations where the DF approach can give large errors in the estimates for K_c and ω_c and the method should therefore not be used 'blindly'. For example, if the process has an unstable FOPDT transfer function, $Ke^{-sTd}/(sT-1)$, the critical frequency is given by

$$\rho = (\tan^{-1}\omega_c T)/\omega_c T \tag{5}$$

and the solution for the exact limit cycle frequency is given by

$$\rho = \log [1 + \tanh (\pi/2\omega_0 T)] \tag{6}$$

An important point is that eqn (5) has solutions for $\rho < 1$ but equation (6) only has solutions for $\rho < 0.693$. Thus estimates for K_c and ω_c deteriorate as T_d/T increases and for this ratio even as low as 0.4 the error in approximating ω_c by ω_0 is around 8%. Similarly describing function analysis applied to a limit cycle in a system with an unstable SOPDT transfer function can result in large errors for some combinations of the parameters [2]. It is possible to do additional relay autotuning tests, with the relay which may include hysteresis, either replacing the controller or being placed in series with it. For example, a tuned filter at the approximate limit cycle frequency can be placed in the loop to obtain an almost sinusoidal limit cycle, so that DF analysis will produce accurate results, or by adding hysteresis to the relay more points on the plant frequency response can be found [3]. Another point worth mentioning is that if the measured critical point values of K_c and ω_c change with the amplitude of the limit cycle, which can be changed by adjustment of the relay heights, this indicates nonlinearity in the process. Nonlinear PID controllers can be designed based on this information to provide responses which do not change significantly with amplitude [4].

Since two measurements are made on the limit cycle if a plant model with two unknown parameters is assumed then these can be estimated from the relay autotuning test if desired, rather than ω_c and K_c. For example, for the

FOPDT plant if K is known then appropriate equations are easily found using DF analysis so that T_d and T can be estimated.

3. Exact Limit Cycle Analysis.

It is possible to evaluate the exact limit cycle waveform given a specific plant transfer function, $G(s)$, using methods based on either of the approaches suggested initially by Tsypkin and Hamel [5]. The exact frequencies have already been given for the limit cycle with an FOPDT plant in the previous section, so this together with the expression for the peak amplitude could be used to calculate exactly T and T_d, assuming K known, if exact measurements of the amplitude and frequency could be obtained. If two such tests, say for a zero and finite hysteresis in the relay were conducted, then four plant parameters could be found. This can also be achieved from a simple asymmetrical limit cycle test if the relay output frequency, on-off ratio and limit cycle positive and negative peak amplitudes are measured [3]. The disadvantages of this approach are the need to solve nonlinear algebraic equations and also that the calculated values can be quite sensitive to measurement errors. However, for those cases where the DF analysis gives very inaccurate results, for example as it would for an unstable FOPDT process with $\rho = 0.55$, it is the only reasonable procedure.

4. Critical Point Design.

The question of how well a controller can be designed based on critical point data was raised in section 3. Although a precise answer is not possible all indications are that if used sensibly the approach can be very successful. For example, if one considers the transfer function

$$G(s) = 320\,(s + 1)/(s + 0.5)(s + 2)(s + 4)(s + 8) \tag{7}$$

then a 'reasonable' reduced order model would appear to be

$$G_r(s) = 35.91/(s^2 + 6.537s + 3.591) \tag{8}$$

since the difference between the unit step responses of the two transfer functions has a maximum error of just over 3% and this discrepancy is such that the step response of $G_r(s)$ would be regarded as a 'good fit' to an experimental (noisy) step response of $G(s)$. However, $G(s)$ has a gain margin of $4.94dB$ and $G_r(s)$ an infinite one. Clearly therefore $G_r(s)$ is not an acceptable reduced model for controller design. On the other hand controllers designed on critical point information usually prove quite robust to errors of up to 10% in ω_c and K_c.

There are several approaches to designing (tuning) a controller given ω_c and K_c for a process. Perhaps one of the simplest and most logical is seen from further consideration of the Z-N approach. Using their suggested tuning parameters for an ideal PID controller $K(1 + (1/sT_i) + sT_d)$ of $K = 0.6K_c$, $T_i = 0.5T_c$ and $T_i = 4T_d$ where $T_c = 2\pi/\omega_c$, it is easy to show that on the compensated frequency response locus the frequency ω_c is moved to the point $0.66\angle -156°$. In general ω_c on the compensated locus can be 'moved around' by varying two controller parameters and as suggested by Z-N this is probably best done by fixing the ratio T_i/T_d and varying K and T_d. It is normally not too difficult to select an appropriate value for the compensated frequency response at the frequency ω_c to give good controller performance if some idea of how the process gain frequency characteristic behaves for frequencies above ω_c since this gives an indication of the gain margin of the compensated system. A model often used for a process plant is the first order plus dead time (FOPDT) transfer function $Ke^{-sT_d}/(1 + sT)$. In [6] PID controller parameters were found for this transfer function to optimise an integral performance criterion for a step input. Formulae were then given to set up these controller parameters based on measurements of ω_c and K_c for this transfer function, assuming K known. Although the formulae were derived for a specific transfer function they were shown to work well for values of ω_c and K_c of other transfer functions.

6. Examples.

Example 1: In this example a plant with a transfer function $G(s) = Ke^{-0.5s}/(s+1)^2$ is considered with $K = 1$. The objective is to show that good tuning can be obtained for a PID controller based on an autotuning test. Using a relay of unit height the measured values of A and ω_0 from a simulation with no noise are $A = 0.287$ and $\omega_0 = 1.90$. Using eqn (1) this gives $\omega_c = 1.90$ and $K_c = 4.44$ which compare with the actual values of 1.92 rads/s and 4.69 for the known plant transfer function. The normalised gain $\kappa = KK_c$ and assuming K has been found by other measurements $\kappa = 4.44$. The tuning formula suggested in [6] which gives optimum parameters for ISTE tuning of a FOPDT plant based on κ, K_c and ω_c, gives $K_p = 2.55$, $T_i = 2.61$ and $T_d = 0.40$. The unit step set point response for these tuning parameters in the controller is shown in Fig. 1 together with that for controller parameters of $K_p = 2.42$, $T_i = 1.99$ and $T_d = 0.63$ obtained by minimisation of the ISTE criterion for the known plant transfer function. The results are very similar considering the approximations involved and the simplicity of the autotuning approach.

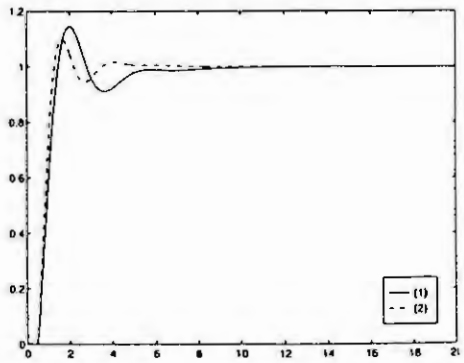


Fig 1 Step response comparison for autotuning (1) and ISTE minimisation (2)

7. Conclusion.

The paper has attempted to show by way of examples that good controller design can be achieved in many instances from a knowledge of the critical point of the process to be controlled. This is only a small amount of model information on the process but it can prove far more important for good controller design than more sophisticated and possibly more accurate models, when based on open loop step response criteria, if they should fail to give a good estimate for the critical points.

8. References.

1. Atherton, D P. 1997. 'Improving the Accuracy of Autotuning Parameter Estimates'. Proceedings, IEEE International Conference on Control Applications, Hartford, USA, pp 51-56
2. Kaya, I and Atherton, D P. 1999. 'Using Limit Cycle Data for Parameter Estimation'. Transactions of the InstMC, Volume 21, No 1, pp 21-29
3. Atherton, D P. 1999. 'PID Controller Tuning'. Computing and Control Engineering Journal, April, pp 175-179
4. Atherton D P, Benouarets M and Nanka-Bruce O. 1993. 'Design of Nonlinear PID Controllers for Nonlinear Plants'. Proceedings, IFAC World Congress '93, Sydney, Australia, Volume 3, pp 355-358
5. Atherton, D P. 1975. Nonlinear Control Engineering – Describing Function Analysis and Design. Van Nostrand Reinhold Co
6. Zhuang M and Atherton D P. 1993. 'Automatic Tuning of Optimum PID Controllers'. IEE Proceedings, Control Theory and Applications, Volume 140, No 3, pp 216-224

MODEL SIMPLIFICATION AND ORDER REDUCTION OF NON-LINEAR SYSTEMS WITH GENETIC ALGORITHMS

Maik Buttelmann and Boris Lohmann
Institute of Automation, University of Bremen
Kufsteiner Straße, D - 28359 Bremen, Germany
E-mail: {mbutt, bl}@iat.uni-bremen.de

Abstract. The simulation, analysis, and controller-design of technical systems are frequently complicated by the order and complexity of the corresponding nonlinear system models. With the order reduction method from [1], [2], systems of lower order (but high complexity) can be calculated. In order to achieve **low order *and* low complexity** this paper presents a genetic algorithm approach which generates complexity constraints to be fulfilled by the order reduction method. This results in models with low order and simple structure with physically interpretable state variables.

System representation, order reduction and model complexity

Starting point of the order reduction [1], [2] is a nonlinear time invariant system

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}), \tag{1}$$

where $\boldsymbol{x}$ is the state vector of dimension n, and $\boldsymbol{u}$ is the input vector of dimension p. An equivalent description of this nonlinear system is:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \boldsymbol{F}\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}). \tag{2}$$

The vector $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u})$ exclusively comprises the nonlinear summands of the elements of $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u})$. Starting from (2) the task of order reduction is to find a system

$$\dot{\tilde{\boldsymbol{x}}}(t) = \tilde{\boldsymbol{f}}(\tilde{\boldsymbol{x}}, \boldsymbol{u}) = \tilde{\boldsymbol{A}}\tilde{\boldsymbol{x}}(t) + \tilde{\boldsymbol{B}}\boldsymbol{u}(t) + \tilde{\boldsymbol{F}}\boldsymbol{g}(\boldsymbol{W}\tilde{\boldsymbol{x}}, \boldsymbol{u}) \tag{3}$$

of lower order $\tilde{n}$ which delivers an approximation of the essential or *dominant* states. These dominant states are chosen by the designer and are combined in the vector $\boldsymbol{x}_{do}$, which is related to the original vector $\boldsymbol{x}$ by

$$\boldsymbol{x}_{do} = \boldsymbol{R}\boldsymbol{x}. \tag{4}$$

Based on the given system (2) and on the user-defined matrix $\boldsymbol{R}$, the method presented in [1], [2] calculates optimal matrices $\boldsymbol{E} = [\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}}, \tilde{\boldsymbol{F}}]$ and $\boldsymbol{W}$ based on system simulations. A disadvantage of this method is the fact that typically all elements of the matrices $\boldsymbol{E}$ and $\boldsymbol{W}$ are non-zero. This corresponds to a high model complexity, since each non-zero element represents one internal coupling within the system.

It is therefore appropriate to not only reduce the system order but also to keep the reduced system simple by aiming at a significant number of zero elements in $\boldsymbol{E}$ and $\boldsymbol{W}$.

In order to achieve this, we first formulate *complexity constraints* on the reduced model (3). These are expressed by *secondary conditions*

$$\boldsymbol{L}_E = \boldsymbol{K}_E\boldsymbol{E}\boldsymbol{H}_E, \qquad \boldsymbol{L}_W = \boldsymbol{K}_W\boldsymbol{W}\boldsymbol{H}_W \tag{5}$$

with prescribed matrices $\boldsymbol{L}$, $\boldsymbol{K}$ and $\boldsymbol{H}$. Conditions of this type can be integrated into the optimization procedure (by first vectorizing and then expressing these equations with the help of the Kronecker Product [1]). For example, the choice $\boldsymbol{H}_E = [1, 0, \ldots, 0]^T$, $\boldsymbol{K}_E = [0, 1, 0, \ldots, 0]$, $\boldsymbol{L}_E = [0]$ forces a zero as first element in the second row of $\boldsymbol{E}$, hence the corresponding element of matrix $\tilde{\boldsymbol{A}}$ becomes $a_{21} = 0$. Details of the reduction method can be found in [1], [2]. For the subsequent considerations it is sufficient to summarize: If the complexity constraints $\boldsymbol{L}$, $\boldsymbol{K}$ and $\boldsymbol{H}$, the original model (2), and the matrix $\boldsymbol{R}$ are given, the reduction method delivers optimal matrices $\boldsymbol{E}$ and $\boldsymbol{W}$ fulfilling the constraints and approximating the behaviour of the original model.

Genetic optimization of complexity constraints

Obviously the choice of appropriate complexity constraints is a crucial task! Because of the huge number of possible choices $\boldsymbol{K}$ and $\boldsymbol{H}$, the use of Genetic algorithms appears to be reasonable. When we assume, that for signifi-

cant parts of the parameter space *small changes* in the complexity constraints cause *small changes* in the approximation quality of the reduced model, we may hope that a Genetic algorithm leads to satisfactory results within reasonable time.

Genetic algorithms are stochastic search methods that mimic the metaphor of natural biological evolution operating on a population of potential solutions applying the principle of survival of the fittest to produce better and better approximations to a solution [3]. At each generation, a new set of approximations is created by the process of selecting individuals according to their level of fitness in the problem domain and breeding them together using operators borrowed from natural genetics. This process leads to the evolution of populations of individuals that are better suited to their environment than the individuals that they were created from, just as in natural adaptation. Genetic algorithms model natural processes, such as selection, recombination, mutation, and insertion. They work on populations of individuals instead of single solutions. In this way the search is performed in a parallel manner.

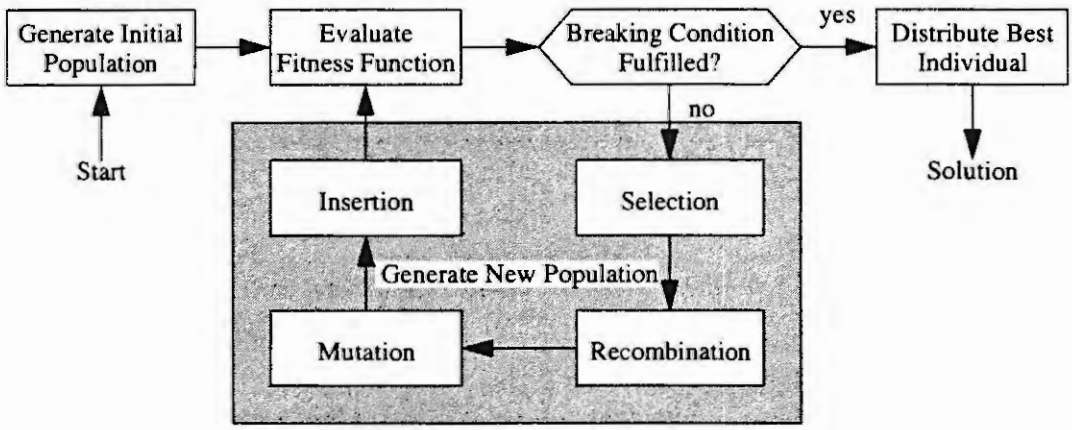


Fig. 1: Structure of the Genetic algorithm

At the beginning of the computation a number of individuals (the initial population) are randomly initialised. The fitness function is then evaluated for these individuals. If the breaking conditions are not fulfilled the creation of a new generation starts. Individuals are selected according to their fitness for the production of offspring. Parents are recombined to produce offspring. All offspring will be mutated with a certain probability. The fitness of each offspring is then computed. The offspring are inserted into the population replacing the parents, producing a new generation. This cycle is performed until the breaking conditions are reached.

Individuals

Every possible choice of $\boldsymbol{K}$ and $\boldsymbol{H}$ is a candidate for the solution of the optimization problem. To use a Genetic algorithm, we must store this solution in an individual. In this case, an individual is build up with ones and zeros. A one signs that a zero is inserted in $\boldsymbol{E}$ or $\boldsymbol{W}$. For example, a desired matrix $\boldsymbol{E}$

$$\boldsymbol{E} = \begin{bmatrix} 0 & x & 0 \\ x & 0 & x \end{bmatrix} \qquad \text{with } x\text{: non-zero element}$$

causes the individual I

$$I = \Big[\underbrace{1\ 0\ 1}_{1^{st}\text{ row of }\boldsymbol{E}}\ \underbrace{0\ 1\ 0}_{2^{nd}\text{ row of }\boldsymbol{E}} \Big] \qquad \text{with 1: zero element in } \boldsymbol{E}\text{, 0: non-zero element in } \boldsymbol{E}$$

The number and the positions of zeros in the matrices $\boldsymbol{E}$ and $\boldsymbol{W}$ are computed by the Genetic algorithm, the non-zero elements are then computed by the reduction method.

Fitness Function

When using a Genetic algorithm, a performance measure is required in order to assess each reduced system model calculated from a set of complexity constraints. This so called "fitness function" F is calculated by simulation of each reduced model with different typical inputs $\boldsymbol{u}(t)$ and by comparison with the behaviour of the original system

(2), rated by the area between the curve of every state of the original system. This area is an indicator for the quality of the approximation the reduced system gives of the original one.

$$F = \underbrace{\frac{\sum_{i=1}^{n} \int_{0}^{\infty} (x_i(t) - w_i \tilde{x}(t))^2 dt}{\sum_{i=1}^{n} \int_{0}^{\infty} x_i^2(t)}}_{\text{model approximation}} - \underbrace{k \cdot \text{sum}(I)}_{\text{model complexity}} \qquad \text{with } w_i \text{ is the } i\text{-th row of } W \tag{6}$$

Optional, the model complexity of the reduced system is also considered in this fitness function. The number of zero-elements in E and W (multiplied with a factor k) is substracted from the value of the model approximation, so that systems with less complexity are ranked better than systems with the same approximation quality and a higher complexity.

Starting population

To start a Genetic algorithm, a population of individuals is needed. The number of ones in an individual is fixed, the positions of the ones in the individual are created randomly. While the Genetic algorithm is working, both the number and the positions are varied.

Sometimes, alternative system models are available from theoretical modelling, e. g. by neglecting some physical effects. The structure of such models, gained from the engineer's know-how, can be formulated as individuals and be added to the starting population.

Genetic operations

Four genetic operations are used for creating offspring of the current population: selection, recombination, mutation, and insertion. Individuals for the recombination are selected depending on their fitness. The recombination produces two offspring of two individuals. A position is selected at random and the bits exchanged between the parents about this point:

parent 1: 0 1 0 | 1 0 0
parent 2: 1 0 1 | 0 0 1
recombination point: 3
offspring 1: 0 1 0 | 0 0 1
offspring 2: 1 0 1 | 1 0 0

This recombination is also called one-point-crossover. After producing the offspring they will be mutated with a low probability. Once an individual is chosen for mutation, a randomly selected position will be changed from one to zero or vice versa.

The new offspring build up the new population. Optional, a less number of the fittest individuals of the old population can replace the worst in the new one.

Breaking conditions

The Genetic algorithm produces new generations with better and better individuals as long as the breaking conditions are not fulfilled. In our implementation, the condition is the number of produced generations.

Best solution

After the Genetic algorithm has stopped producing new generations, the best one is proposed as the solution.

Example

The algorithm is tested on the system shown in fig. 2. It is a test bed for a combustion engine linked on an eddy current break with a flexible shaft. The break stator is linked with a spring damper unit to the foundations.

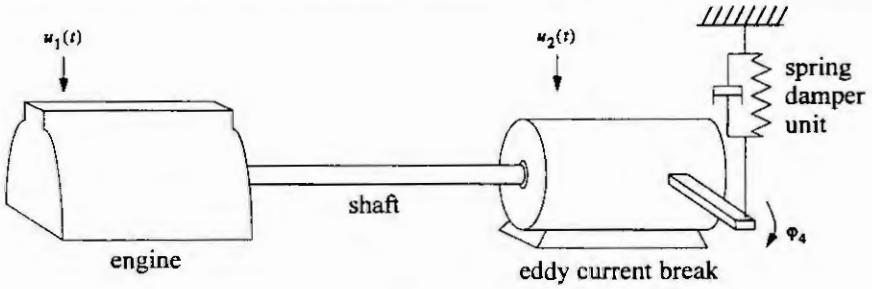


Fig. 2: Test bed

For comparison, the system is modelled in two ways:

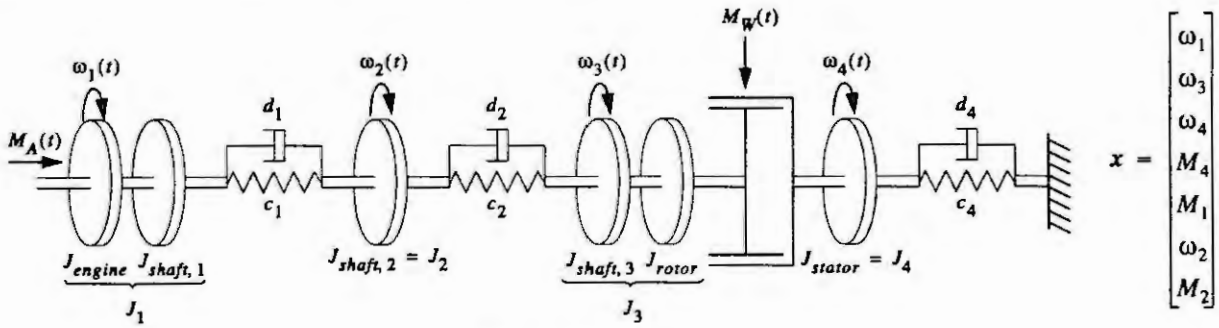


Fig. 3: Model 7th order

The first model is shown in fig. 3 (see [1] for parameters). It is 7th order and is used as the reference model in the fitness function and for the order reduction. A model of 3rd order can be found without the order reduction method when the shaft is modelled with only a flywheel without any spring damper unit instead of the complex modelling with three flywheels and two spring damper units. The corresponding model is shown in fig. 4.

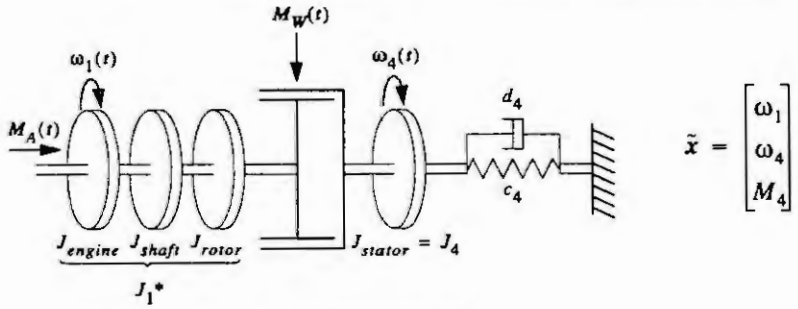


Fig. 4: Model 3rd order

The matrices

$$\tilde{A} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -2 \\ 0 & 2000 & -100 \end{bmatrix} \quad \tilde{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad \tilde{F} = \begin{bmatrix} 6{,}135 & -6{,}135 \\ 0 & 2 \\ 0 & 100 \end{bmatrix} \tag{7}$$

are describing the model of 3rd order with 14 zeros in $\tilde{A}, \tilde{B}, \tilde{F}$.

Starting the order reduction method without any secondary conditions delivers also a model of 3rd order, but with a higher complexity (5 zeros in $\tilde{A}, \tilde{B}, \tilde{F}$):

$$\tilde{A} = \begin{bmatrix} 0 & -44 & -33{,}3 \\ 0 & -0{,}1 & -2 \\ 0 & 2000 & -100 \end{bmatrix} \quad \tilde{B} = \begin{bmatrix} -2 & -20 \\ 0 & -0{,}1 \\ -0{,}4 & -2{,}2 \end{bmatrix} \quad \tilde{F} = \begin{bmatrix} 6 & -2{,}6 \\ 0 & 2 \\ -0{,}1 & 100 \end{bmatrix} \tag{8}$$

The order reduction with secondary conditions and genetic minimization of the model complexity computes the following result[1]:

$$\tilde{A} = \begin{bmatrix} 0 & 0 & -33{,}3 \\ 0 & 0 & 0 \\ 0 & 0 & -100 \end{bmatrix} \quad \tilde{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad \tilde{F} = \begin{bmatrix} 6 & -2{,}7 \\ 0 & 0 \\ 0 & 100 \end{bmatrix} \tag{9}$$

The matrices $\tilde{A}, \tilde{B}, \tilde{F}$ contains 16 zeros, so this model is of lowest complexity and is only 2nd order, since the second state can be removed.

The simulations of the three reduced models compute the following values for the approximation quality:

	Approximation	**Complexity**	**Fitness F**
Model 3rd order, manual reduced	303	14 zeros in $\boldsymbol{E}$	$303 - (1 \cdot 14) = 289$
Model 3rd order, computed	210	5 zeros in $\boldsymbol{E}$	$210 - (1 \cdot 5) = 205$
Model 2nd order, computed	100	16 zeros in $\boldsymbol{E}$	$100 - (1 \cdot 16) = 84$

Table 1: Overall view of the fitness F of the reduced models, $k = 1$

Beside the good approximation quality, the model 2nd order can also be interpreted physically: The state $\tilde{x}_2$ of the reduced system is the rotation of the stator. In steady state, the rotation is zero – the value the order reduction computes for $\tilde{x}_2$. Simulation proves that this simplification, found by the Genetic algorithm, is acceptable!

Summary

The order reduction method computes a model of a nonlinear system with reduced order but high complexity. The Genetic algorithm produces secondary conditions without any knowledge about the possibilities of complexity reduction. With these secondary conditions, a significant number of zero elements is aimed in $\boldsymbol{E}$ and $\boldsymbol{W}$ and a model of reduced order *and* low complexity is computed. Simulations of an illustrative example show that the secondary conditions produced with the Genetic algorithm provide understandable results. The complexity is much lower and the model approximation is better than without the secondary conditions.

Possibilities of also simplifying $\boldsymbol{W}$ are to be studied, as well as the behaviour when treating technical systems of higher order.

References

1. Lohmann, B.: Ordnungsreduktion und Dominanzanalyse nichtlinearer Systeme. VDI-Fortschrittsberichte, Reihe 8, VDI-Verlag, Düsseldorf, 1994.
2. Lohmann, B.: Order Reduction and Determination of Dominant State Variables of Non-linear Systems.In: Proc. of the IMACS Symposium on Mathematical Modelling, 1st MATHMOD, Vienna 1994, 239 – 243 and Mathematical Modelling of Systems, Vol. 1, (1995), 77 – 90, (extended version).
3. Documentation of the Genetic and Evolutionary Algorithm Toolbox for use with Matlab (GEATbx). TU Illmenau.
4. Schöneburg, E, et al.: Genetische Algorithmen und Evolutionsstrategien. Addison-Wesley, 1994.

1 Only the matrix $\boldsymbol{E}$ is considered. The complexity of $\boldsymbol{W}$ is not reduced.

MODELLING THE MOTIONS OF A FAST FERRY WITH THE HELP OF GENETIC ALGORITHMS

B. de Andrés Toro, S. Esteban, J.M. Giron-Sierra, J.M. de la Cruz
Dept. ACYA. Fac. Fisicas. Universidad Complutense de Madrid
Ciudad Universitaria. 28040 Madrid. Spain. E-mail: deandres@eucmax.sim.ucm.es

Abstract. Models suitable for control analysis are found for a particular fast-ferry. A replica of the ship has been employed to get experimental data. Another set of data has been obtained with the program PRECAL. The models are transfer functions. Genetic algorithms have been employed to explore the best data adjustment, for a set of candidate models. Satisfactory results have been obtained for 20, 30 and 40 knots.

Introduction.

Fast ships are of increasing importance. But there are several problems that arise against speed: for instance, those related with the vertical motions of the ship (heaving and pitching motions) due to ocean waves. Passenger comfort is degraded by vertical motions. To counteract the effects of waves, our ship has two flaps below the transom, that we can move under control. Our problem is to move the flaps in an adequate way: this is a matter of control design. From the point of view of automatic control, is most convenient to have a good model of the plant to be controlled. And what is more, the model should be suitable for mathematical analysis. This is the objective of the present work.

If possible, the good way to obtain a model is by first principles: in this case, to reason with the physics of a moving ship. The scientific literature helps in this sense. The main phenomena are studied in the books [1,2,3]. Focusing on the dynamic response to waves, the article [4] analyse the case of regular waves, and [5] completes the view for more general situations. In addition, the article [6] presents several curves describing the behaviour of some fast ships.

Since linear transfer functions are very convenient for automatic control study, the modelling effort has been directed to such objective. This has been not easy. But, by means of a vast exploration, with the help of *genetic algorithms*, satisfactory results have been obtained.

The Problem.

The research deals with the ship "Silvia Ana", a fast-ferry working now in La Plata (south summer) and in the Baltic Sea (north summer). References [7,8] contain technical descriptions of the ship. To increase speed, the ship is aluminium-made. The main characteristics of the ship are the following: 110m. length, 14.696m. beam, 2.405m. draught, 475tons. deadweight , 1250 passengers. Our work contemplates the heaving and pitching motions, which take place in the vertical plane. The ship is moving, with a speed V_b, against head waves of pulsation w_{ola}. The pulsation of encounter is: $w_e = w_{ola} + w_{ola}^2 . V_b / g$

Both the measured inputs (the waves) and the outputs (heaving and pitching motions), can be processed to obtain an approximation of the frequency response of the ship (as required by the transfer function approach), related to the pulsation of encounter. Once the frequency responses are plotted, the problem to be solved is to get two transfer functions in agreement with the data: one for heaving motion, and the other for pitching motion.

The Data.

To obtain the pertinent data, a scaled down replica of the ship has been built, and employed for experiments in a towing tank institution (CEHIPAR, Madrid, Spain). By means of a wave generator, in a big pool, experimental data have been measured, with the replica, for speeds of 20, 30 and 40 knots, and for regular and irregular waves. The experiments with regular waves have been performed for 15 different wavelengths. The irregular waves have been generated according with STANAG 4194 (Standarized Wave and Wind Environments and Shipboard Reporting of Sea Conditions), for sea states 4, 5 and 6, with JONSWAP spectra. The data have been sampled, and saved as computer files. From this information, some identification studies have been done in terms of time-series [9], and a pre-conditioning of the data has been accomplished.

The mentioned institution (CEHIPAR) can also provide with simulated data, generated by the program PRECAL, which uses a geometrical model of the ship to predict her dynamic behaviour. With this program, a complete set of data tables have been generated, reproducing the same conditions of the experiments with regular waves. The data generated by PRECAL can be displayed as Bode diagrams, representing the frequency response of the ship. Figures 1 and 2 show the Bode diagrams for heaving and pitching motions, at 30 knots.

From the Bode diagrams, and reasoning about the physics of the ship motions, some clues and criteria can be extracted for modelling in the form of transfer functions. For instance, at low frequencies the transfer function of heaving must have unity gain, $|G(jw)|_{w\to 0} = 1$

and the transfer function of pitching must have zero gain and a phase of 90 °: $|G(jw)|_{w\to 0} = 0$ $|G(jw() = 90^{o}$

Modelling with the Help of Genetic Algorithms.

Starting from the acquired experience, and from the study of experimental data, the following general expression of the transfer functions of heaving and pitching can be written:

$$G(s) = \frac{Heave(s)}{Wave(s)} = \frac{k\cdot(s^2+s\cdot p_o+p_1)(s^2+s\cdot p_2+p_3).....}{(s+p_4)\cdot(s^2+s\cdot p_5+p_6)\cdot(s^2+s\cdot p_7+p_8).....} = \frac{k_1.s^m+k_2.s^{m-1}+k_3.s^{m-3}+.....+k_{m+1}}{r_1.s^n+r_2.s^{n-1}+r_3.s^{n-3}+.....r_{n+1}}$$

$$G(s) = \frac{Pitch(s)}{Wave(s)} = \frac{k\cdot s.(s^2+s\cdot p_o+p_1)(s^2+s\cdot p_2+p_3).....}{(s+p_4)\cdot(s^2+s\cdot p_5+p_6)\cdot(s^2+s\cdot p_7+p_8).....} = \frac{k_1.s^m+k_2.s^{m-1}+k_3.s^{m-2}+.....+k_{m+1}.s}{r_1.s^n+r_2.s^{n-1}+r_3.s^{n-2}+.....+r_{n+1}}$$

These transfer functions must agree, as much as possible, with the experimental data (the Bode diagrams). Our problem is to determine the best combination of m and n, and the values of the numerator and denominator coefficients, for such purpose. To measure how good is the agreement, the following adjustment criterion is defined:

$$J_{Real} = \sum_{i=2}^{n}(Real_{Data} - Real_{Model})^2 \qquad J_{Imag} = \sum_{i=2}^{n}(Imag_{Data} - Imag_{Model})^2 \qquad J_I = J_{Real} + J_{Imag} \qquad J = \frac{100}{J_I}$$

The data of the Bode diagrams are complex numbers. After adding squared errors, the above equations get a final value J: the higher J, the better the adjustment.

At this point, our main idea is to define a set of combinations of m and n values, and apply genetic algorithms for each combination to determine the value of the numerator and denominator coefficients that gets the best J. This idea has been implemented in two stages: first an exploration of value ranges for the coefficients (here to have some clues from physics are important), and second a complete study getting the best J for each valid combination of m and n, with m and n taking values from 2 to 7 (always m <= n).

Specifications of the Genetic Algorithm.

There are many books and articles explaining genetic algorithms. For instance [10,11]. A very important application field where genetic algorithms prove to be useful, is optimisation. In particular, the adjustment problem (between model and experimental data) is an optimisation problem.

The key for the application of genetic algorithms to a problem, is to be able to represent the problem in terms of chromosomes and a fitting function. In our case, the fitting function will be the J of the adjustment. Regarding to chromosomes, one of the important features of genetic algorithms is that they allow to incorporate the knowledge of the problem to the code. We take advantage of that in our codification. Instead of looking directly for the values of the numerator and denominator coefficients, we will look for their roots, in order to reduce the searching space. The coefficients can have values belonging to [-∞,+∞], but, as we know the plant is stable, the value of the denominator roots must belong to [-∞, 0], so we have eliminated a half of the searching space. Moreover, since the module of the roots must be of the same order than the maximum pulsation of encounter, the range of values must be inside [-4,0]. With that, we have delimited a reasonable searching space. Similar reasoning can be applied to the numerator roots, with the only difference that they can be positive.

Suppose that the combination m=4, n=5, has been selected. In that case, we have to find the roots of the following functions:

$(s^2 + p_0 s + p_1)(s^2 + p_2 s + p_3)$ in [-4,4]

$(s + p_4)$, $(s^2 + p_5 s + p_6)$, and $(s^2 + p_7 s + p_8)$ in [-4,0]

The roots of these polynomials are complex numbers. We will denote them as z_1+z_2j, z_3+z_4j,d_1+d_2j,d_3+d_4j, and d_5+d_6j respectively. An individual will be represented by a chromosome with 10 genes, as follows:

$$[z_1\ z_2\ z_3\ z_4\ d_1\ d_2\ d_3\ d_4\ d_5\ d_6\ ]$$

A direct codification by integer numbers has been chosen, to improve implementation performance. Thus, for a precision equal to 0.001, the alphabet is Ω = [-4000,4000], where 4000 represents 4.000, 3999 represents 3.999, and so on. For instance, if we have a solution represented by the following individual:

[243 –1325 1019 –2681 2386 0 –2912 –1283 –601]

the roots of the numerator and denominator are: 2.43±1.325j, 1.019±2.681j, 2.386±0j,-2.419±2962j, and 1.283±601j, and the model represented by the chromosome is:

$$G(s) = \frac{(s^2 - 0.486 \cdot s + 1.8147)(s^2 - 2.038 \cdot s + 8.2261)}{(s + 2.386)(s^2 + 4.838 \cdot s + 14.3313)(s^2 + 2.565 \cdot s + 2.0073)}$$

Taking advantage of our previous experience with genetic algorithms, the specification of the genetic operators has been the following: probability of mutation: 0.008; probability of crossover: 0.8; initial population: 10 individuals; parent selection by roulette-wheel, 4 substitutions/generation.

Each evolution encompass 10000 generations, along 40 epochs. A superindividual is created, by local optimisation, at the end of each epoch.

Results.

The following table shows part of the results obtained for heaving and pitching. For instance, the entry w3p1416 means 30 knots, pitching, 1 real zero, 4 complex zeroes, 1 real pole, 6 complex poles. Each case (entry) included in the table means a study with genetic algorithms, repeating five times a complete evolution of 10000 generations.

Model	Zeroes	Poles	J
w2h0406	(0.562 ± 1559j) (-1.086 ± 0.045j)	(-0.384 ± 0.556j) (-2.797 ± 0.706j) (-0.368 ± 1.085j)	7033.31
w3h0406	(0.732 ± 2.026j) (-0.636 ± 2.959j)	(-3.032 ± 3.977j) (-0.489 ± 0.855j) (-0.286 ± 1.514j)	6202.35
w2p1214	(2.048 ± 0j) (0.482 ± 1.824j)	(-4.784 ± 0j) (-0.256 ± 1.360j) (-0.347 ± 0.817j)	1008.93
w3p1416	(4.01 ± 0j) (-0.704 ± 3.023j) (1.029 ± 3.018j)	(-6.009 ± 0j) (-0.276 ± 1.708j) (-0.582 ± 1.202j) (-3.465 ± -0.145j)	2090.48

From the results obtained, the best transfer functions can be selected for 20, 30 and 40 knots. For instance, at 30 knots, the best models are the following:

$$G(s) = \frac{\text{Heave}(s)}{\text{Wave}(s)} = \frac{16.52s^5 + 7.518s^4 + 75.75s^3 + 95.37s^2 + 0.466s + 301}{s^6 + 23.27s^5 + 163.1s^4 + 353.6s^3 + 605s^2 + 543s + 301}$$

$$G(s) = \frac{\text{Pitch}(s)}{\text{Wave}(s)} = \frac{s^6 - 4.148s^5 + 11.89s^4 - 48.22s^3 + 60.17s^2 - 192s}{s^7 + 13.11s^6 + 60.09s^5 + 159.9s^4 + 282.1s^3 + 374.9s^2 + 284.2s + 162.3}$$

Figures 1 and 2 show how good is the agreement between data and models .

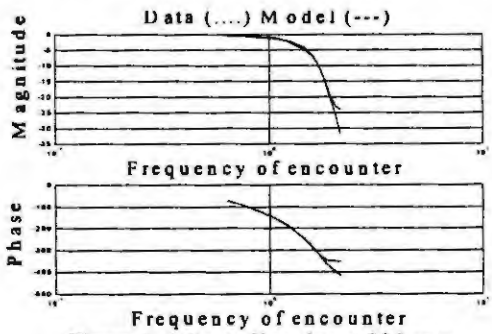


Figure 1: Adjust of heaving at 30 knots

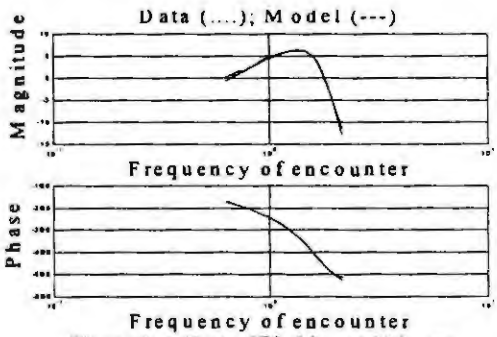


Figure 2: Adjust of Pitching at 30 knots

Once the best models have been found, we can validate them by a comparison between the motions measured by CEHIPAR with the replica, and the motions predicted by the model. Figures 3 and 4 show the results at 30 knots for regular waves. Figures 5 and 6 for irregular waves, also at 30 knots.

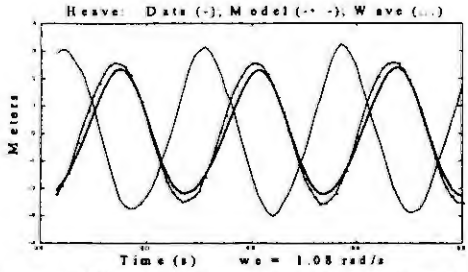


Figure 3: Validation of Heaving with Regular Waves

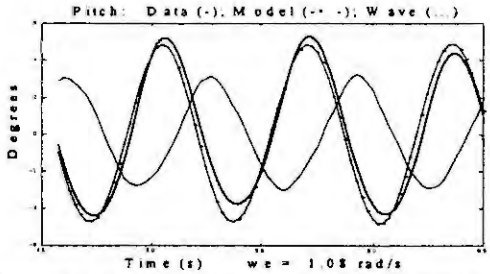


Figure 4: Validation of Pitching with Regular Waves

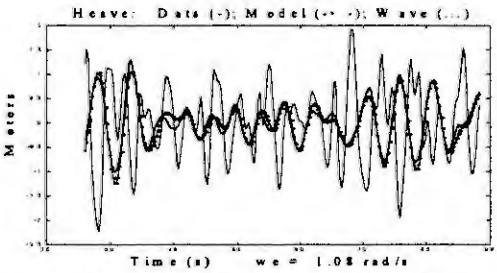


Figure 5: Validation of Heaving with Irregular Waves

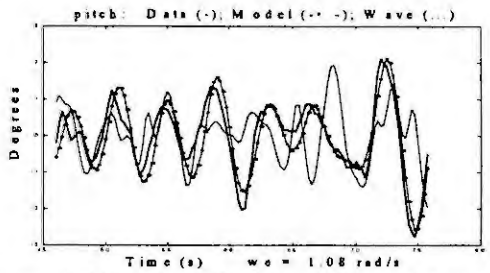


Figure 6: Validation of Pitching with Irregular Waves

Conclusions.

This paper dealt with the modelling, for control purposes, of the vertical motions of a fast ferry. The study started with an extensive experimental work, to obtain relevant data.

By means of genetic algorithms, a large set of candidate models have been explored. The results allow us to select good models for heaving and pitching motions at 20, 30 and 40 knots, with heading seas. The transfer functions obtained are more complicated than the simplifications published in the scientific literature (for conventional ships). Since the set of evolutions is large, the genetic algorithm has been parallelized [12] for calculation speed up. We think the method established is conceptually simple, and that can be easily applied to other modelling problems. In the future, we plan to model other sea conditions (not only heading waves) and other motions of the ship.

Acknowledgments: The authors want to thank the support of the CICYT Spanish Committee (project TAP97-0607-C03-01), and the collaboration of the CEHIPAR staff.

References.

1. Lewis, E.V., Principles of Naval Architecture. SNAME, New Jersey, 1989.
2. Lloyd, A.R.J.M., Seakeeping: Ship Behavior in Rough Weather. Ellis Horwood, John Wiley, New York, 1988.
3. Fossen, T.I., Guidance and Control of Ocean Vehicles. John Wiley, New York, 1994.
4. Korvin-Kroukovski, B.V. and Jacobs, W.R., Pitching and Heaving Motions of a Ship in Regular Waves. SNAME T., 65 (1957), 590-632.
5. Salvesen, N., Tuck, E.O. and Faltinsen, O., Ship Motions and Sea Loads. SNAME T., 78 (1970), 250-285.
6. Van Sluijs, and Gie, T.S., Behaviour and Performance of Compact Frigates in Head Seas. Intl. Shipbuilding Progress, 19, 210 (1972), 35-52.
7. Anonymous, 126 m Long Spanish Fast Ferry Launched. Fast Ferries, September (1996), 19-20.
8. Anonymous, Silvia Ana: Results of First Year's Service. Ship & Boat Intl., Jan/Feb (1998), 15-16.
9. De la Cruz, J.M., Aranda, J., Ruiperez, P., Diaz, J.M., and Maron, A., Identification of the Vertical Plane Motion Model of a High Speed Craft By Model Testing in Irregular Waves. IFAC Conf. CAMS'98, Fukuoka, (1998).
10. Michalewicz, Z., Genetic Algorithms + Data Structures = Evolution Programs. Springer Verlag, 1996.
11. Goldberg, D.E., Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley, 1989.
12. De Andrés, B., Hidalgo, J.I., Prieto, M., Lanchares, J. and Tirado, F., A Parallel Genetic Algorithm for Solving the Partitioning Problem in Multi-FPGA Systems. In: Proc. 3rd. Intl. Meeting of Vector and Parallel Processing, Porto , (1998), 717-722.

HIERARCHICAL-DECENTRALIZED SOLUTIONS OF SUPERVISORY CONTROL

S.CHAFIK and E.NIEL
Laboratoire d'Automatique Industrielle LAI, INSA de Lyon
20, Avenue Albert Einstein, 69621 Villeurbanne cedex, France
Tel: (33)-4-72-43-81-98 Fax: (33)-4-72-43-85-35 e-mail : {schafik, eniel}@lai.insa-lyon.fr

Abstract. The major problem of the supervisory control is the extension states of the system. This paper describes a method that handle state explosion in large systems by combining both hierarchical (vertical) and decentralized (horizontal) supervisory control concepts. In this case we show that the normality of decentralized supervisor of the low level (of the hierarchical structure) is preserved at the abstraction high level (hierarchical High level). So, the non conflicting (if it is checked at the low level) is also preserved at the high level without disturbing the hierarchical consistency of the proposed structure

I- Introduction

Different researches propose solutions that solve the complexity problem of supervisory control of discrete event systems (DES) by offering an optimal legal behavior of a system. Among these researches we quote the modular supervision of Ramadge and Wonham [6], the decentralized supervision of Lin and Wonham [2][3] and the hierarchical supervision of Zhong and Wonham [9][10].
The application of the decentralized supervision is based on the achievement of a control task which requires only the management of a local events subset Σ_{loc} of the global event labels Σ. At the global level, the legal behavior K of G is obtained from local supervisors S_{loc}. K will be then optimal.
The hierarchical supervision considers two levels of hierarchy and consists of obtaining a simplified and abstracted models. The efficiency of this approach depends on the hierarchical consistency of the two levels. Once hierarchical consistency has been achieved for the bottom level and the first level up, say (G_0, G_1), the hierarchical constructions may be repeated on bringing in a next higher level G_2. However in order to bring in other higher hierarchical levels G_i, we propose the hierarchical-decentralized supervisory control concept which, allows to enrich the theory of supervision of DES already developed by Ramadge and Wonham then expanded by many works. Our idea consists of combining concepts of hierarchical and decentralized supervisory controls in order to limit the extension of the system to other hierarchical levels, then to reduce the risk of the state explosion. This concept consists of dividing the high level model of the process to several small subprocesses. Each one of these subprocesses possesses its own local supervisor. we show that the presence of the decentralized supervisors in the hierarchical structure does not disturb the hierarchical consistency of the two levels and then the legal behavior at each level is optimal.

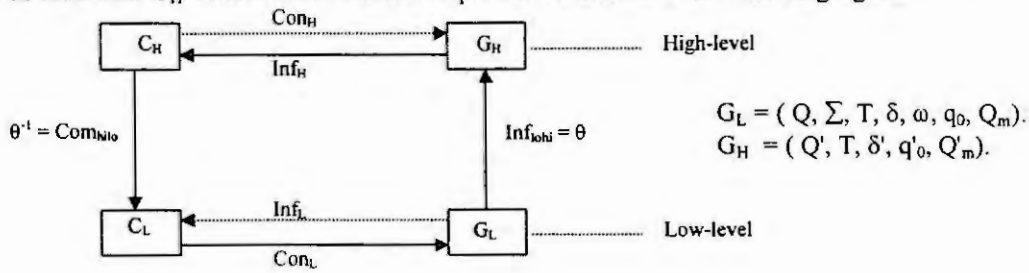
II - Decentralized approach

The decentralized approach consists of decomposing the system into many sub-systems and then the global specification will typically be a conjunction of a number of component specifications [2] [3] [5] [6].
In this case the accomplishment of a given controlled sub-task may only need the management of certain subsets of the event alphabet Σ, hence the interest of this architecture where each subsystem or process Gi observes and controls only event symbols belonging to the corresponding Σ_i. For this, [2, 4] have introduced the natural projection notion P_i witch, consists simply of erasing events that do not belong to Σ_i.
A local supervisor S_{iloc} controls only events in Σ_{iloc} according to $L(G_i) = P_i(L(G))$ and synthesize (for G_i) the language $K_{iloc} = SupC_{iloc}(P_i(L(G)) \cap E_{iloc}) \subset \Sigma_i^*$, the global effect of S_{iloc} ($\widetilde{S}_i$) synthesis $L(G)) \cap P_i^{-1} K_{iloc}$
Now let S_{1loc}, S_{2loc}, ..., S_{nloc} n local supervisors witch, control individually G. the Concurrent action of their global supervisors is modeled by the conjunction of $\widetilde{S}_1, \widetilde{S}_2, ..., \widetilde{S}_n$ .that is $L(\underset{i\in I}{\wedge} \widetilde{S}_i, G) = SupC(L(G) \cap E)$ and $E = \underset{i\in I}{I} P_i^{-1} E_{iloc}$

$L(\underset{i\in I}{\wedge} \widetilde{S}_i, G)$ synthesizes the same language than those of centralized supervisor[4]. For extra information about the decentralized supervisory control see [1] [2] [3] [5] [6].

III - Hierarchical approach

This approach is based on obtaining simplified models of the process. The model of Zhong [7] [8] [9] [10] considers two levels of hierarchy consisting of a low-level plant $(G_L)^1$ and its controller C_L along with a high-level model G_H (abstract model of G_L) and its controller C_H. These two levels are coupled as the shown in the following figure.

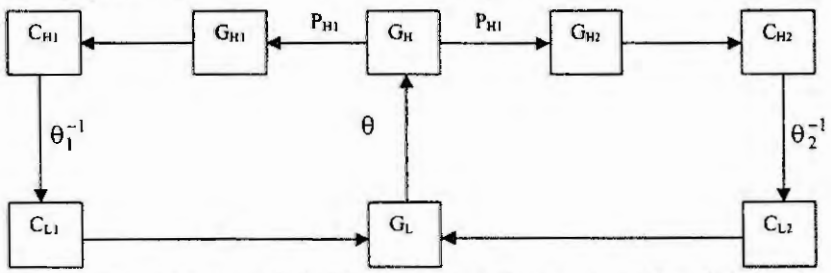


We say that the low-level hierarchical consistency is achieved when $L(C_L, G_L) = (E_L^{\uparrow})^2 = (\theta^{-1}(E_H))^{\uparrow}$. The pair ($G_L$, G_H) will be said to possess hierarchical consistency when $E_L^{\uparrow} = (\theta^{-1}(E_H))^{\uparrow} = E_L$, then $\theta(\theta^{-1}(E_H))^{\uparrow} = E_H$.

VI- Hierarchical-decentralized supervisory control

The concept of hierarchical-decentralized supervisory control is based on two fundamental theories : the decentralized and hierarchical supervisory controls. The hierarchical-decentralized supervision combines these two concepts and presents a structure, which is divided vertically and horizontally while checking consistency, optimality and normality notions for these two levels. This approach is based on obtaining simplified local models. It consists of dividing the system into two hierarchical levels:

- the low level consists of a model G_L of the plant and two or several local controllers (supervisors) C_{Li}.
- the high level contains an abstract model of G_L and two or several local submodels. Each submodel is supervised by a local manager (supervisor) C_{Hi}. The two levels are coupled as shown in the following figure :



G_L represents the Moore generator model of the plant and G_H represents the abstract hierarchical model of G_L. Let $G_{Hi} = P_{Hi}(G_H)$ be a local model of G_H obtained by the projection of G_H over an alphabet T_i. We suppose that :

(1) local models are not necessarily disjoints: $\Sigma \cap \Sigma \neq \varnothing$ or $\Sigma \cap \Sigma = \varnothing$

(2) G_L is an SOCC, that is $T = T_c \cup T_u$ and G_L does not contain any partners;

(3) $K_{Hi} = L(C_{Hi}, G_H) \cap P_{Hi}^{-1} P_{Hi}(K_{Hi})$. Then $K_{Hi} = \widetilde{K}_{Hi}$ with $\widetilde{K}_{Hi} = L(G_H) \cap P_{Hi}^{-1}(K_{Hi,loc})$ and $K_{Hi,loc} = L(C_{Hiloc}, G_{Hiloc})$; since $K_{Hi} = \widetilde{K}_{Hi}$, then G_H is locally controllable;

(4) $G_{Lm} = G_L$ then $\theta(L_{Lm}) = L_{Hm}$ and $\theta^{-1}(L_{Hm}) = L_{Lm}$.

---

[1] G_L is a Moore automaton where Σ,T,δ and ω are respectively the input and output event sets and the input and output transition maps of G_L.

[2] $K_L^{\uparrow}$ Represents the supremal controllable language of a language K

our aim is to equip local models G_{Hi} with local managers(supervisors), then to find the local appropriate low level controllers. Let C_{Hi} be a local manager for G_H. C_{Hi} is defined by a map

$C_{Hi} : L_{Hi} \times T_{ci} \rightarrow \{0,1\}$ such that $C_{Hi}(t_i, \tau) = 1 \ \forall \ t_i \in L_{Hi}$ and $\tau \in T_{uc}$ or $C_{Hi}(t_i, \tau) = 0$ if $\tau \in T_{ic}$ and t must be disabled

Let Δ_H be the set of all high level events to be disabled following the generation of any string t of L_H and define Δ_{Li} the set of local low level events to be disabled following the generation of a string s_i of $P_{Li}(L_L)$ and t_i of L_{Hi}, by a map

$\Delta_{Li} : L_{Li} \times L_{Hi} \rightarrow Pwr(\Sigma_{ci})$ with $\Sigma_{ci} = \Sigma_c \cap \Sigma_i$

$\Delta_{Li}(s_i, t_i) = \{\sigma \in \Sigma_{ci} / (\exists\ s)\ s\sigma \in L(G_L)\ \&\ P_{Li}(s) = s_i\ \&\ \theta(s) = t\ \&\ P_{Hi}(t) = t_i\ \&\ \omega(s\sigma) \in \Delta_H(t)\ \&\ \omega(P_{Li}(s\sigma)) \in \Delta_{Hi}(t_i)\}$.

When the hierarchical loop is closed through θ, the application of θ on the string $s \in L(G_L)$ leads to a high-level string $t = \theta(s) \in L_H$ such that $P_{Li}(s) = s_i$ and $\theta(s_i) = t_i$. Then the control of C_{Li} on G_L will be defined by a map

$$C_{Li}[P_{Li}(s), \theta(P_{Li}(s)), \sigma] = \begin{cases} 0 \text{ if } \sigma \in \Delta_{Li}[P_{Li}(s), \theta(P_{Li}(s)] \\ 1 \quad \text{otherwise} \end{cases}$$

According to the definition of C_{Li} we note that when $\sigma \in \Sigma_{ci}$, σ can be authorized or disabled by C_{Li}, but when $\sigma \in \Sigma - \Sigma_{ci}$, will be it authorized or disabled ? The following theorem replies to this question while being based the theorem.3.2.

Theorem 4.1 :Let $L_j \subset L(G)$.Then $\forall\ j \neq i$, $L_j \subset L(C_i, G)$. ❑

For proofs of Theorem 4.1 and the following propositions see [1].

Now, suppose that $K_{Hi,loc}$ is nonempty, closed and controllable sublanguage such that $K'_{Li,loc} = \theta^{-1}(K_{Hi,loc})$. $K'_{Li,loc}$ may be not controllable. In this case it will be then necessary to determine its corresponding supremal controllable sublanguage which is given by $K'^{\uparrow}_{Li,loc} = [\theta^{-1}(K_{Hi,loc})]^{\uparrow} = K_{Li,loc}$. Moreover knowing that K_{Hi} is normal (from the assumption.3), then $K_{Hi} = \widetilde{K}_{Hi}$ and G_H is locally controllable with respect to the family of local sublanguages $\{E_{Hi}\}$. However, when passing at the low level, is normality preserved? this is, $\theta^{-1}(K_{Hi}) = K'_{Li}$, is it normal? and is G_L locally controllable?

Proposition.4.1 : Suppose that K_{Hi} is normal with respect to $(L(G_H), P_{Hi})$, then $K'_{Li} = \theta^{-1}(K_{Hi})$ and $K'^{\uparrow}_{Li} = K_{Li}$ are also normal with respect to $(L(G_L), P_{Li})$ ❑

Suppose that the global specification of G_L is given by the language $E_L = \prod_{i=1}^{n} P_{Li}^{-1} E_{Li,loc} \subset \Sigma^*$ with $E_{Li,loc} \subset \Sigma_i^*$.

When acting alone, a controller C_{Li} synthesizes the optimal sublanguage $K_{Li} = SupC[L(G_L) \cap P_{Li}^{-1} E_{Liloc}]$. If K_{Hi} is normal, then the corresponding low level K_{Li} is also normal (proposition.4.1). In this case, we can determine $\widetilde{K}_{Li} = L(\widetilde{C}_{Li}, G_L)$ by the following proposition

Proposition.4.2: Let $\widetilde{K}_{Hi} = L(G_H) \cap P_{Hi}^{-1}(K_{Hi,loc})$, then $\widetilde{K}_{Li} = (\theta^{-1}(\widetilde{K}_{Hi}))^{\uparrow} = K'^{\uparrow}_{Li} = L(\widetilde{C}_{Li}, G_L)$ ❑

When all controllers C_{Li} act together , we can determine through the proposition.5.3 that the concurrent action of all C_{Li} synthesis $\underset{i \in I}{\wedge} \widetilde{K}_{Li} = L(\underset{i \in I}{\wedge} \widetilde{C}_{Li}, G_L)$ with $\widetilde{K}_{Li}$ representing the global effect of $K_{Li,loc}$.

Proposition.4.3 : Let $\widetilde{K}_H = \underset{i \in I}{\wedge} \widetilde{K}_{Hi} = L(G_H) \cap \prod_{i=1}^{n} P_{Hi}^{-1} K_{Hi,loc}$ and $K'_L = \underset{i \in I}{\wedge} K'_{Li} = \theta^{-1}(\widetilde{K}_H)$, then

$$\widetilde{K}_L = L(\underset{i \in i}{\wedge} \widetilde{C}_{Li}, G_L) = \underset{i \in I}{\wedge} \widetilde{K}_{Li} = \underset{i \in I}{\wedge} K'^{\uparrow}_{Li} \quad \text{with } \widetilde{K}_L = L(G_L) \cap \prod_{i=1}^{n} P_{Li}^{-1} K_{Li,loc}$$ ❑

The proposition.4.3 guarantees the low-level hierarchical consistency even in the presence of the concurrent action of all local controllers C_{Li}. The behavior of the corresponding high level satisfies the constraint

$$\theta(L(\underset{i \in I}{\wedge} \widetilde{C}_{Li}, G_L)) \subseteq \widetilde{K}_H . \qquad (4.1)$$

If the equality of (4.1) holds, G_L and G_H will be said to possess hierarchical consistency. The hierarchical consistency of (G_L, G_H) will be achieved when $\theta(\widetilde{K}_L) = \widetilde{K}_H$, but before checking this equality, we must check first that when C_{Li} acts alone, we obtain $\theta(\widetilde{K}_{Li}) = \widetilde{K}_{Hi}$.

Proposition.4.4: Suppose that G_L is an SOCC. Let $\tilde{K}_{Hi}$ be nonempty, closed and controllable with $\tilde{K}_{Hi} = L(G_H) \cap P_{Hi}^{-1}(K_{Hi,loc})$ and $\tilde{K}_{Li} = (\theta^{-1}(\tilde{K}_{Hi}))^{\uparrow}$. Then, $\theta(\tilde{K}_{Li}) = \tilde{K}_{Hi}$ ❑

When C_{Li} acts alone on G_L the equality $\theta(\tilde{K}_{Li}) = \tilde{K}_{Hi}$ is validated. However, when all local supervisors act simultaneously on G_L, Is this equality always validated? namely $\theta(\underset{i\in I}{\wedge} \tilde{K}_{Li}) \overset{?}{=} \underset{i\in I}{\wedge} \tilde{K}_{Hi}$ (4.2)

Proposition.4.5 : Suppose that G_L is an SOCC and let $\tilde{K}_H = \underset{i\in I}{\wedge} \tilde{K}_{Hi} \subseteq L(G_H)$ with K_H is nonempty closed and controllable and $\tilde{K}_{Li} = (\theta^{-1}(\tilde{K}_{Hi}))^{\uparrow}$. Then $\theta(\tilde{K}_L) = \tilde{K}_H$ with $\tilde{K}_L = \underset{i\in I}{\wedge} \tilde{K}_{Li}$ ❑

According to the proposition .4.5, the hierarchical consistency of (G_L, G_H) will be achieved when $\theta(\tilde{K}_L) = \tilde{K}_H$

V- Conclusion

In general the model of a production process could be very large, containing a great number of states, the corresponding supervisor also could be complex. The hierarchical-decentralized concept aims to reduce the complexity of supervisors by injecting decentralized supervisors at the high level. The formal contribution of this approach has enabled us to check that the presence of the decentralized supervision in a hierarchical structure does not modify the hierarchical consistency notion and the obtained decentralized supervisors of the different levels are optimal. The normality of languages is conserved in the hierarchical structure

References :

1. Chafik.,S and Niel,E., The Extension of Hierarchical Supervision to Hierarchical-Decentralized Supervision. Submitted to European journal of control. 1999

2. Lin, F.and Wonham,.W.M., Decentralized Supervisory Control of Discrete-Event Systems. In: Inform.Sci. 1988. vol.44. N° 3. pp 199-244.

3. Lin, F.and Wonham,.W.M., Decentralized Control and Coordination of Discrete-Event Systems with Partial Observation. IEEE Transactions on Automatic Control.. 1990. vol.35. N° 12. 1330-1337.

4. Ramadge, P.J. and Wonham, W.M., Supervisory Control of has Class of Discreet Vent processes. SIAM Journal on Control, and Optimization. January 1987. vol.25. N°.25. 206-230

5. Ramadge, P.J. and Wonham, W.M., The Control Of Discreet Event Systems. IEEE Transactions one Automatic Control. January 1989. vol 77. N°.1. 81-98

6. Thistle, J.P., Supervisory Control Of Discrete Event Systems. Mathl Comput Modelling. vol.23. N°.11/12. 1996. 25-53

7. Wong, K.C. and Wonham,.W.M., Hierarchical of Discrete-Event Systems. Discrete Event Dynamic Systems. 1996. vol 6. 241-273

8. Wonham,W.M., Notes on Control of Discrete-Event Systems. System Control Group, Dept of Electrical & Computer Engineering. University of Toronto. 1999

9. Zhong, H and Wonham,.W.M., On the Consistency of Hierarchical Supervision in DES. IEEE Transactions on Automatic Control. 1990. vol 35. N° 10. 1125-1134.

10. Zhong, H., Hierarchical Control Of Discrete-Event Systems. Ph.D. thesis, Department of Electrical Engineering, University of Toronto. 1992, also appears as technical Report 9208, Systems Control Group, Department of Electrical Engineering, university of Toronto, July,1992

MODELLING AND COMPENSATION OF REPEATABLE RUNOUT IN HARD DISK DRIVE[1]

Feng Zheng†[2], Jian-Xin Xu‡, Tong Heng Lee‡, Tao Zhu‡, Paul M. Frank†
† Department of Measurement and Control, FB9
Duisburg University, D-47048 Duisburg, Germany
‡Department of Electrical Engineering
National University of Singapore, Singapore 119260

Abstract. A new method to attenuate repeatable runout (RRO) is presented in this paper. In our scheme, notch filters are used to extract the harmonics to be attenuated from the output signal and then feed the extracted signals back into the input to reduce the harmonics. The effectiveness of the proposed method was verified by experimental results. An advantage of our approach is that we can design the frequency response of the filters according to the characteristics of RRO. Another advantage is that only the phase characteristics of the plant is needed in the proposed approach.

1 Introduction

Periodic disturbances are inherent in any rotating machinery. In disk data storage technology, the periodic disturbances appear in the position error of the hard disk head following a data track. This repeatable runout (RRO) in the position of the hard disk head with respect to the track center can be a considerable source of tracking error. Hence some control effort is usually taken to compensate for at least some of these periodic disturbances.

RRO compensators can be broadly classified into two classes: internal model based ones and external model based ones [3]. The most widely used internal model based compensator is the Q-filter algorithm developed by Chew et al. [2] and Tsao et al. [6]. An important class of external model based compensators (EMBC) is adaptive feedforward cancellation (AFC) [1, 5], by which selective runout harmonics can be cancelled. In this paper a new method to compensate RRO is proposed. In our scheme, we use notch filters to extract the harmonics to be cancelled from the output signal and then feed the extracted signals back into the input to attenuate the harmonics.

2 RRO Compensation Scheme

The basic idea for the compensation of only one harmonic is illustrated in Fig. 1: where $P(z)$ and $H(z)$

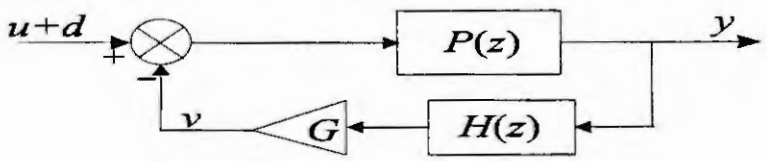


Figure 1: Basic structure for compensation of one harmonic

are the transfer functions of the plant and notch filter in z-domain, respectively, G is the gain of the amplifier, u is the control command signal, y is the position error signal, and d represents disturbance. Notice that the dynamics of the hard disk drive, servo, observer and tracking-following controller is included in $P(z)$. It is obtained from Fig. 1 that

$$Y(e^{j\omega}) = \frac{P(e^{j\omega})}{1 + G\,H(e^{j\omega})P(e^{j\omega})}(U(e^{j\omega}) + D(e^{j\omega})), \tag{1}$$

where $Y(e^{j\omega})$, $U(e^{j\omega})$ and $D(e^{j\omega})$ denote the discrete Fourier transforms of y, u and d, respectively.

Assumption 1 *The frequency spectrums of the disturbance and control command signal are separated:*

$$|U(e^{j\omega})| = 0 \qquad \forall\omega \in \omega_{dsupp} \stackrel{\text{def}}{=} \{\omega \in [0,\pi] : |D(e^{j\omega})| > 0\}.$$

[1]This work was supported partly by Alexander von Humboldt Foundation.
[2]Corresponding author. Email: zhengf@unidui.uni-duisburg.de

Remark 1 *The condition implied in Assumption 1 is too strict. From engineering point of view it is sufficient to define separability as follows:*

$$|U(e^{j\omega})| < \epsilon_1 \qquad \forall \omega \in \omega_{dsupp\epsilon}, \stackrel{\text{def}}{=} \{\omega \in [0,\pi] : |D(e^{j\omega})| > \epsilon_2\},$$

where ϵ_1 and ϵ_2 are small positive numbers.

The ideal frequency response of notch filter is as follows:

$$|H_{id}(e^{j\omega})| = \begin{cases} 1 & \text{when} \quad \omega = \omega_0 \\ 0 & \text{otherwise,} \end{cases} \tag{2}$$

where ω_0 represents the frequency of harmonic to be compensated. Suppose that $|G\, H_{id}(e^{j\omega_0})P(e^{j\omega_0})| = |G\, P(e^{j\omega_0})| \gg 1$. From Assumption 1 we have

$$|Y_{id}(e^{j\omega})| \doteq \begin{cases} \frac{1}{|G|} \cdot |D(e^{j\omega_0})| & \text{when} \quad \omega = \omega_0 \\ |P(e^{j\omega})| \cdot |U(e^{j\omega})| & \text{otherwise.} \end{cases} \tag{3}$$

It is shown by equation (3) that in ideal case the amplitude of the relevant harmonic of the disturbance is reduced to $\frac{1}{|G|}$ times its original value, while the useful control signal remains invariant.

One kind of practical notch filters to approximate ideal notch filter (2) is as follows [4]:

$$H(z) = \frac{1-\sin\theta_2}{2} \frac{1-z^{-2}}{1+\sin\theta_1(1+\sin\theta_2)z^{-1}+\sin\theta_2 z^{-2}}, \tag{4}$$

where θ_1 and θ_2 are two parameters relevant to the central frequency and bandwidth of the filter.

Practical notch filter (4) is a narrow bandwidth filter. Let ω_c and B denotes the central frequency and -3 dB attenuation bandwidth of the notch filter. Then we have [4]:

$$\omega_c = \theta_1 + \frac{\pi}{2} \qquad \text{for} \quad |\theta_1| < \frac{\pi}{2}; \qquad B = 2\arctan\frac{1-\sin\theta_2}{1+\sin\theta_2}.$$

It is shown that the central frequency and bandwidth of the notch filter can be adjusted independently by regulating the parameters θ_1 and θ_2, respectively. This fact facilitate the design of the filter.

To overcome the roundoff noise accumulation in the state vector loop of the notch filter, its lattice form realization is used, which is realized by the following recursive form:

$$\begin{aligned} &v_1(k) = \cos\theta_2\, y(k) - \sin\theta_2\, x_2(k), \quad v_2(k) = \sin\theta_2\, y(k) + \cos\theta_2\, x_2(k), \\ &x_1(k+1) = \cos\theta_1\, v_1(k) - \sin\theta_1\, x_1(k), \quad x_2(k+1) = \sin\theta_1\, v_1(k) + \cos\theta_1\, x_1(k), \\ &v(k) = \frac{G}{2}[y(k) - v_2(k)], \end{aligned}$$

where x_1 and x_2 are the state variables of the notch filter in the above realization, and y(k) and v(k) are the input and output of the filter. There initial values are set as $x_1(0) = x_2(0) = 0$.

The multiple harmonics compensation scheme is illustrated in Fig. 2

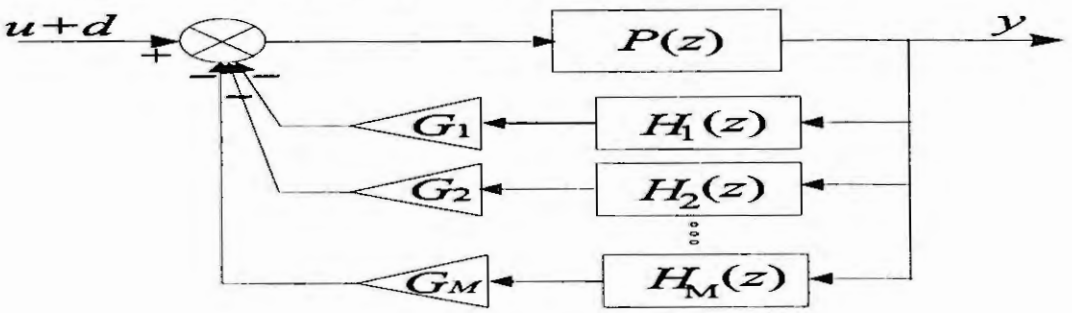


Figure 2: Compensation scheme for multiple harmonics

where

$$H_i(z) = \frac{1-\sin\theta_{i2}}{2} \frac{1-z^{-2}}{1+\sin\theta_{i1}(1+\sin\theta_{i2})z^{-1}+\sin\theta_{i2}z^{-2}},$$

and it is supposed that $\omega_{ci} \neq \omega_{cj}$, for $i \neq j$, where ω_{ci} is the central frequency of the filter H_i.

If all the $H_i(z)$, $i = 1, \ldots, M$, are ideal notch filters and suppose Assumption 1 holds, we have

$$|Y(e^{j\omega})| \doteq \begin{cases} \frac{1}{|G_i|} \cdot |D(e^{j\omega_{ci}})| & \text{when} \quad \omega = \omega_{ci},\ i = 1, \ldots, M \\ |P(e^{j\omega})| \cdot |U(e^{j\omega})| & \text{otherwise,} \end{cases} \tag{5}$$

where it is supposed that $|G_i\ H_i(e^{j\omega_{ci}})P(e^{j\omega_{ci}})| = |G_i\ P(e^{j\omega_{ci}})| \gg 1$.

It is shown again by equation (5) that the amplitude of the relevant harmonic of the disturbance is reduced to $\frac{1}{|G_i|}$ times its original value, while the useful control signal remains invariant

3 Experimental Results

Our experiments were made on a test stand, which mainly consists of a 3.5 inch form-factor magnetic disk drive, a 16-bit fixed point DSP plus peripheral circuit, position sensing circuit, and a host PC. Seeking and tracking controllers are already built in the test stand. The position error signal is read directly from the position signal demodulator. The disk drive rotates at about 55.4 Hz. The sampling rate is 3436.4 Hz.

Fig. 3 shows the frequency response of the test stand, which is identified by the experimental data.

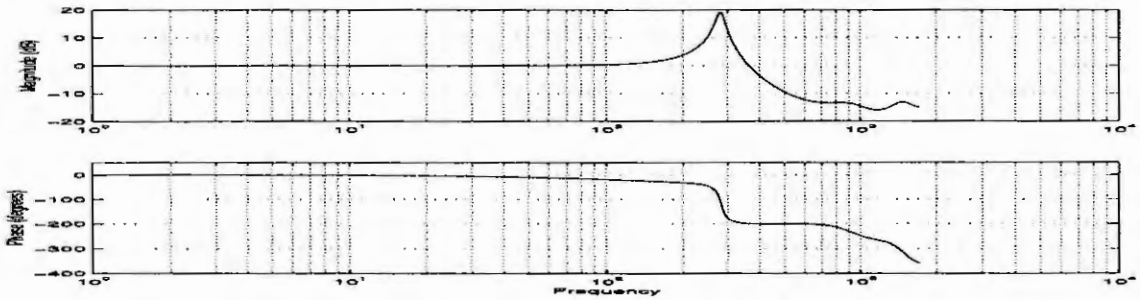


Figure 3: Estimated frequency response of the test stand

First, we applied four filters to the 1st, 2nd, 3rd, and 4th harmonics. Secondly, we applied four filters to the 7th, 8th, 9th, and 10th harmonics. The corresponding parameters of the notch filters are as follows:

- central frequencies: $\theta_{11} = -1.4695$, $\theta_{21} = -1.3682$, $\theta_{31} = -1.2669$, $\theta_{41} = -1.1656$, $\theta_{71} = -0.8617$, $\theta_{81} = -0.7604$, $\theta_{91} = -0.6591$, and $\theta_{10,1} = -0.5579$;
- frequency bandwidth: $\theta_{i2} = 0.49\pi$, $i = 1, 2, 3, 4, 7, 8, 9, 10$;
- gains: $G_1 = G_2 = G_3 = G_4 = 10$, $G_7 = G_8 = G_9 = G_{10} = -30$.

Notice that the gains of 7th, 8th, 9th and 10th notch filters are set to be negative. This is due to the fact that the phase of the loop transfer function is almost shifted 180° relative to the input, which is shown by Fig. 3. The experimental results are illustrated in Figs. 4 and 5.

The power spectrum densities (PSD) of the PESs are shown in Fig. 4. It turns out that the corresponding harmonics is greatly attenuated. It is also observed that other harmonics are not amplified by this method, which is not the case for some other methods [1].

Note that there is a big resonance at the frequency of 290Hz in our test stand. The cause is unknown. Due to this resonance, the effect of compensation for the 5th harmonics is not clear.

The PESs in time domain are shown in Fig. 5. Due to the existence of the big resonance at the frequency of 290Hz and other harmonics which is not compensated, it is not easy to determine convergence rate. However, we can roughly see that the system undergoes track seeking mode before $t = 0.02$ seconds, and it is in transient phase between $t = 0.02$ seconds and $t = 0.04$ seconds. That is to say the convergence time of the notch filters is approximately 0.02 seconds.

4 Concluding Remarks

In this paper a new method to attenuate RRO is presented. The effectiveness has been verified by the experimental results. An advantage of our approach is that we can design the frequency response of the

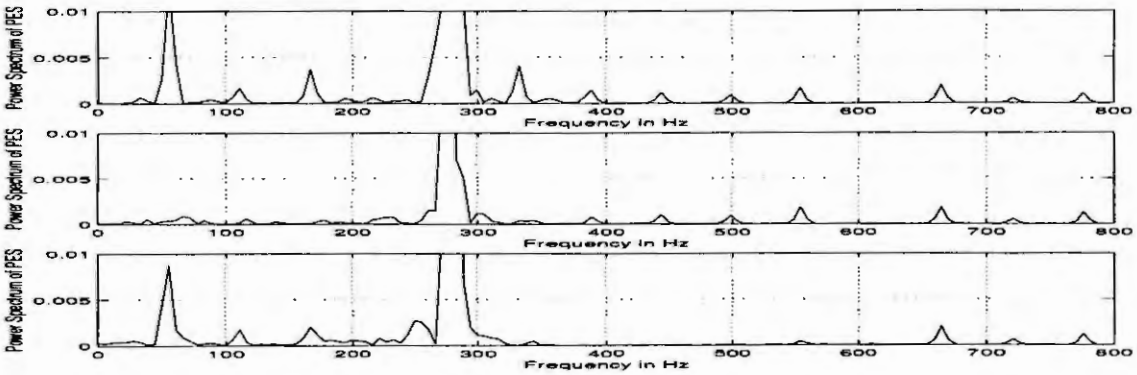


Figure 4: Power spectrum density of PES. (a) with no compensation; (b) with compensation notch filters applied to harmonics 1, 2, 3, 4; (c) with compensation notch filters applied to harmonics 7, 8, 9, 10.

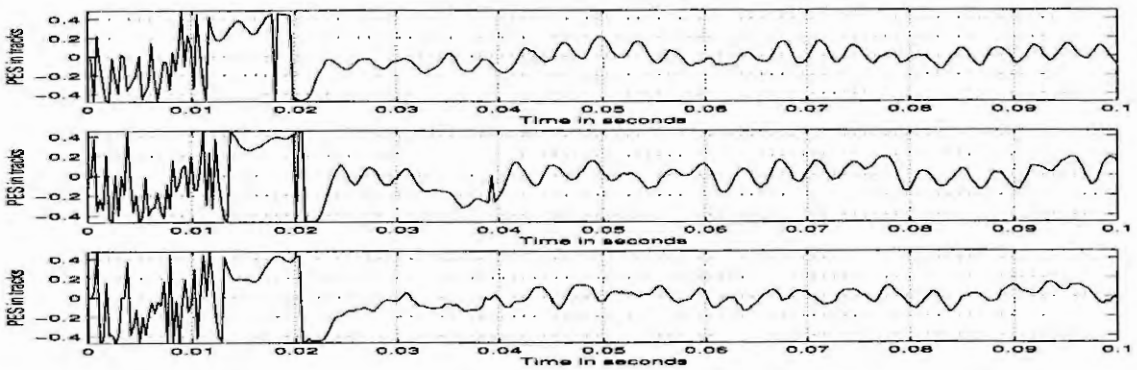


Figure 5: Time domain response. (a) with no compensation; (b) with compensation notch filters applied to harmonics 1, 2, 3, 4; (c) with compensation notch filters applied to harmonics 7, 8, 9, 10.

filter according to the characteristics of RRO since the two parameters of notch filter have direct relations to the parameters of the harmonic to be attenuated. Specifically, we can first determine θ_1 according to the rotation frequency of the disk drive. Then we can determine θ_2 according to the range of the variation in the rotation frequency of the disk drive if any. By a suitable choice of θ_2, we can achieve robustness in some degree against small variations of the harmonic frequencies. Another advantage is that exact model of the plant is not needed in the proposed approach.

It is worthy to note that other harmonics are not amplified by this method.

Acknowledgment: We would like to acknowledge Dr. Tony Huang and Dr. Tony Huang and Dr. Guoxiao Guo of Data Storage Institute, Singapore for their fruitful help in our experiment.

References

[1] Bodson, M., Sacks, A. and Khosla, P., Harmonic generation in adaptive feedforward cancellation schemes, *IEEE Trans. on Automatic Control*, vol.39, pp.1939-1944, 1994.

[2] Chew, K. and Tomizuka, M., Digital control of repetitive errors in disk-drive systems, *IEEE Control Systems Magazine*, vol.10, pp.16-20, Jan. 1990.

[3] Kempf, C., Messner, W., Tomizuka, M. and Horowitz, R., Comparison of four discrete-time repetitive control algorithms, *IEEE Control Systems Magazine*, vol.13, pp.48-54, Dec. 1993.

[4] Regalia, P. A., *Adaptive IIR Filtering in Signal Processing and Control*, Marcel Dekker, Inc., New York, 1995.

[5] Sacks, A. H., Bodson, M. and Messner, W., Advanced Methods for Repeatable Runout Compensation, *IEEE Trans. on Magnetics*, vol.31, no.2, pp.1031-1036, 1995.

[6] Tsao, T. and Tomizuka, M., Adaptive and repetitive digital control algorithms for noncircular matching, *Proc. of American Control Conf.*, vol.1, Atlanta, 1988.

MODELLING A CLASS OF NONLINEAR PLANTS AS LPV-SYSTEMS VIA NONLINEAR STATE TRANSFORMATION

S. Sommer and U. Korn
Institute of Automation, Otto-von-Guericke-University Magdeburg
PF 4120, 39016 Magdeburg / Germany

Abstract. We present a new approach to model a class of nonlinear systems as LPV-systems via a nonlinear state transformation. In the sequel this reduces the controller design for input-affine nonlinear plants to *linear parameter-varying systems* (LPV-systems), where the varying parameters are functions of the state variables. This practice allows us to use LPV-controller design methods. Not yet solved is the right choice of a linear state dependent like structure. The mentioned nonlinear state transformation enables a more systematical way without an intuitive action. In this paper we design a static output feedback controller with pre-compensator that ensures *Quadratic Stability*. The proposed conception will be demonstrated on a bio reactor model.

Introduction

One limitation of controller design methods for nonlinear plants via local linearization is the aspect that the controller guarantees stability only close to a single operating point. Gain scheduling can extend the validity of the linearization technique to a range of operating points but there are stability problems in the case of fast changes of the input signal.

Another technique is based on reducing controller design for input-affine nonlinear plants to *linear parameter-varying systems* (LPV-systems), where the varying parameters are functions of the state variables $p(t) = p(x(t))$. The strategy can be found in many publications, for instance [6, 7] and will be named often as *quasi-LPV approach*. This practice allows us to use efficient LPV-controller design methods for nonlinear control systems with guaranty of stability for a set of possible state variables $x(t) \in \mathcal{M}_x$ respectively a class of nonlinear plants. Given the following nonlinear input-affine SISO-system:

$$\dot{x}(t) = f(x(t)) + g(x(t))\,u(t), \quad y(t) = h(x(t)). \tag{1}$$

First step is the choice of a state dependent representation of the nonlinear system (1):

$$\dot{x}(t) = A(x(t))\,x(t) + B(x(t))\,u(t), \quad y(t) = C(x(t))\,x(t). \tag{2}$$

The parameter functions $p_i(x(t))$ replace all nonlinearities of the state space matrices in (2) and will be collected to the state dependent vector $p(x(t)) = \begin{pmatrix} p_1(x) & \cdots & p_m(x) \end{pmatrix}^T$. In further statements we abbreviate $p(x(t))$ with $p(x)$. Now we obtain

$$\dot{x}(t) = A(p(x))\,x(t) + B(p(x))\,u(t), \quad y(t) = C(p(x))\,x(t) \tag{3}$$

and define it as a *formal linear parameter-varying system* (FLPV-system), also called as *quasi-LPV system*. Both, LPV-system and FLPV-system have the same general properties. The difference between both forms consists in the state dependence of the parameter vector $p(x)$ in FLPV-systems.

One problem in this strategy is choosing a suitable state dependent structure (2) and defining the parameter functions $p_i(x)$, because there exists a infinite number of feasibilities to built a FLPV-representation (3) from the nonlinear plant (1). Therefore we suggest a new approach for a more methodical way, that realizes a nonlinear state transformation which converts the nonlinear plant (1) into a FLPV-system (3) with a special normal form. Thereby, the controller design will be done with a FLPV-system that always has the same structure.

The paper is organized as follow. In the next section we introduce the nonlinear state transformation and derive the algorithms for computing the transformation vector and parameter functions. It follows the presentation of a controller design method for LPV-systems. Afterwards the transformation in connection with controller design will be demonstrated on a bio reactor model. The paper concludes with a summary.

Nonlinear State Transformation

The main topic of our contribution is a new approach to model a class of nonlinear plants as *linear parameter-varying systems* via the nonlinear state transformation:

$$z = T(x) = \begin{pmatrix} T_1(x) & \cdots & T_{n-1}(x) & T_n(x) \end{pmatrix}^T. \tag{4}$$

Our goal is the conversion of the nonlinear plant (1) into a FLPV-system (3) by using the state transformation (4). Firstly, we want to receive a state dependent form (2). With the new state vector (4) we get:

$$\dot{z} = A(x)\,z + B(x)\,u = A(x)\,T(x) + B(x)\,u. \tag{5}$$

The derivative of the state vector (4) in combination with the original nonlinear plant (1) results:

$$\dot{z} = \frac{\partial T(x)}{\partial x}\dot{x} = \frac{\partial T(x)}{\partial x}\left[f(x) + g(x)\,u\right]. \tag{6}$$

By equating (5) and (6) we obtain:

$$\frac{\partial T(x)}{\partial x}\left[f(x) + g(x)\,u\right] = A(x)\,T(x) + B(x)\,u. \tag{7}$$

The equation (7) will be split into:

$$\frac{\partial T(x)}{\partial x} f(x) = A(x)\,T(x) \quad \text{and} \quad \frac{\partial T(x)}{\partial x} g(x) = B(x). \tag{8}$$

With the special structure of the matrices

$$A(p(x)) = \begin{pmatrix} 0 & 1 & 0 & 0 \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 \\ p_{A1}(x) & \cdots & \cdots & p_{An}(x) \end{pmatrix}, \; B(p(x)) = \begin{pmatrix} p_{B1}(x) \\ \vdots \\ p_{Bn}(x) \end{pmatrix} \tag{9}$$

equations (8) simplifies to

$$\begin{aligned} \tfrac{\partial T_1(x)}{\partial x} f(x) &= T_2(x) \\ &\vdots \\ \tfrac{\partial T_{n-1}(x)}{\partial x} f(x) &= T_n(x) \\ \tfrac{\partial T_n(x)}{\partial x} f(x) &= p_{A1}(x)T_1(x) + \cdots + p_{An}(x)T_n(x) \end{aligned} \quad \text{and} \quad \begin{aligned} \tfrac{\partial T_1(x)}{\partial x} g(x) &= p_{B1}(x) \\ &\vdots \\ \tfrac{\partial T_n(x)}{\partial x} g(x) &= p_{Bn}(x). \end{aligned} \tag{10}$$

The differential equations (10) form the algorithm for computing of the new state variables $T_2(x), \ldots, T_n(x)$ and the state dependent parameter functions $p_{B1}(x), \ldots, p_{Bn}(x)$ directly and $f(x) = (f_1(x) \cdots f_n(x))^T$ allows the computation of the parameters $p_{A1}(x), \ldots, p_{An}(x)$:

$$p_{A1}(x) = \frac{\frac{\partial T_n(x)}{\partial x_1} f_1(x)}{T_1(x)}, \quad p_{A2}(x) = \frac{\frac{\partial T_n(x)}{\partial x_2} f_2(x)}{T_2(x)}, \quad \cdots \quad , \quad p_{An}(x) = \frac{\frac{\partial T_n(x)}{\partial x_n} f_n(x)}{T_n(x)}. \tag{11}$$

The use of $z_1 = T_1(x) = h(x)$ results in a constant output matrix of the transformed system:

$$C = \begin{pmatrix} 1 & 0 & \cdots & 0 \end{pmatrix}. \tag{12}$$

This is an advantage for the application of output feedback control, because the outputs of original nonlinear system and transformed system are identical. Furthermore, it is a necessary requirement that the state dependent parameter vector $p(x)$ is bounded

$$p(x) = \begin{pmatrix} p_{A1}(x) & \cdots & p_{An}(x), & p_{B1}(x) & \cdots & p_{Bn}(x) \end{pmatrix}^T \in \left[\underline{p},\, \overline{p}\right] \tag{13}$$

for all possible values of the state vector $x(t) \in \mathcal{M}_x$. With (9), (12) and (13) we obtain a FLPV-System which can be used for LPV-controller design. Thereby, two advantages exist. An affine parameter dependent representation of the matrices $A(p(x))$ and $B(p(x))$ is achieved. Additionally we have a normal form representation of the matrices $A(p(x))$, $B(p(x))$ and C. The controller synthesis is always put down to a FLPV-System with the same configuration.

Controller Design

In connection with controller design we regard the FLPV-system as a LPV-system. That means, design a controller for the system (3) respectively for the transformed system (9), (12) which stabilizes the closed loop for all $p(x) \epsilon [\underline{p}, \overline{p}]$. Afterwards, the received controller can be applied to the original nonlinear plant (1). For design a state feedback law

$$u(t) = -K\,x(t)$$

the following problem was adapted from [5].

Find a matrix $Y \epsilon \Re^{p \times n}$ *and a matrix* $W \epsilon \Re^{n \times n}$ $(W = W^T > 0)$ *such that*

$$A(p)\,W + W\,A^T(p) - B(p)\,Y - Y^T\,B^T(p) < 0$$

for all $p \epsilon \mathcal{P}$, *then the state feedback controller*

$$u(t) = -Y\,W^{-1}\,x(t)$$

stabilizes the LPV-system:

$$\dot{x}(t) = A(p(t))\,x(t) + B(p(t))\,u(t).$$

This LMI-condition is based on *Quadratic Stability* [1] of a LPV-system. In combination with the transformed system (9), (12) we have to use $u(t) = -K\,z(t)$. If $A(p)$ and $B(p)$ are affine functions of the parameters p_i $(i = 1, \ldots, m)$ and the parameters $p_i \epsilon [\underline{p}_i, \overline{p}_i]$ are varying in the parameter set $\mathcal{P} := \{\Pi_i\}_{i=1}^{r=2^m}$ with the corners Π_i then the stability test can be reduced to a finite number of LMI's:

$$A(\Pi_i)\,W + W\,A^T(\Pi_i) - B(\Pi_i)\,Y - Y^T\,B^T(\Pi_i) < 0, \quad W = W^T > 0 \quad i = 1, \ldots, r. \tag{14}$$

Let there are m parameters, then the parameter set $\mathcal{P}$ has $r = 2^m$ corners Π_i. The corners result from the parameter bounds $\underline{p}_i$ and $\overline{p}_i$. LMI conditions for output feedback could be fouud in [3]. A lot of design procedures for LPV-systems respectively FLPV-systems has been developed in the last years, such as [1, 4, 6, 7].

Example

The proposed transformation in connection with controller design will be demonstrated and applied to a bio reactor model [2]

$$f(x) = \begin{pmatrix} \frac{\mu_{max}\,x_1\,x_2}{k_m + x_2 + k_1\,x_2^2} \\ -\frac{1}{Y}\frac{\mu_{max}\,x_1\,x_2}{k_m + x_2 + k_1\,x_2^2} \end{pmatrix}, \; g(x) = \begin{pmatrix} -x_1 \\ x_2 - x_{2f} \end{pmatrix}, \; h(x) = x_2 \tag{15}$$

with the biomass concentration $x_1(t)$, substrate concentration $x_2(t)$ and dilution rate as input $u(t)$. The characteristic values are $x_{2f} = 4\,g/l$, $Y = 0.4$, $k_m = 0.12\,g/l$, $k_1 = 0.4545\,l/g$, $\mu_{max} = 0.53 \cdot 1/h$, $x_1(0) = 1.48\,g/l$, $x_2(0) = 0.3\,g/l$. A reasonable working range will be assumed as $x_1 \epsilon [1.48, 1.568]$ and $x_2 \epsilon [0.08, 0.3]$. The control objective is the reduction of the substrate concentration to $x_{2s} = 0.1$. The transformed system will be computed with (10) and (11). By choosing $T_1(x) = h(x) = x_2$ and inserting the working range we obtain:

$$\dot{z}(t) = \begin{pmatrix} 0 & 1 \\ p_{A1}(x) & p_{A2}(x) \end{pmatrix} z(t) + \begin{pmatrix} p_{B1}(x) \\ p_{B2}(x) \end{pmatrix} u(t), \quad y(t) = \begin{pmatrix} 1 & 0 \end{pmatrix} z(t), \tag{16}$$

$$z_1 = T_1(x) = x_2, \quad z_2 = T_2(x) = -\frac{\mu_{max}\,x_1\,x_2}{Y(k_m + x_2 + k_1\,x_2^2)},$$

$$p_{A1}(x) = -\frac{\mu_{max}^2\,x_1\,x_2}{Y(k_m + x_2 + k_1\,x_2^2)^2} \epsilon [-2.18, -1.47],$$

$$p_{A2}(x) = \frac{\mu_{max}\,x_1(k_1\,x_2^2 - k_m)}{Y(k_m + x_2 + k_1\,x_2^2)^2} \epsilon [-5.91, -0.73],$$

$$p_{B1}(x) = (x_{2f} - x2) \epsilon [3.7, 3.92],$$

$$p_{B2}(x) = \frac{\mu_{max}\,x_1(2\,k_m\,x_2 + x_2^2 - k_m\,x_{2f} + k_1\,x_2^2\,x_{2f})}{Y(k_m + x_2 + k_1\,x_2^2)^2} \epsilon [-22.34, -1.4251],$$

$$p(x) = \begin{pmatrix} p_{A1}(x) & p_{A2}(x) & p_{B1}(x) & p_{B2}(x) \end{pmatrix}^T \epsilon\, [\underline{p},\, \overline{p}]. \tag{17}$$

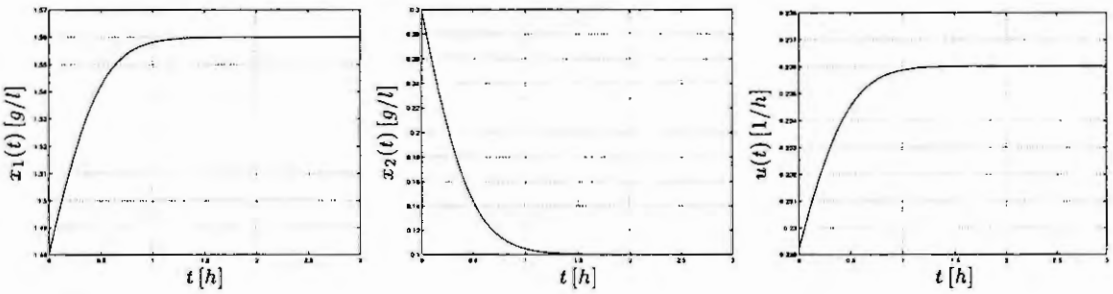
The resulting transformed system will be used for a controller design. We have to find a controller $u(t) = -K\, z(t)$ that stabilizes the system (16) for all $p(x)\,\epsilon\,[\underline{p},\, \overline{p}]$ (17). Due to four parameter functions, there are $r = 2^4 = 16$ corners of the parameter space hence a set of $r = 16$ LMI's (14). One possible solution that satisfies the set of 16 LMI's was calculated:

$$K = Y\, W^{-1} = \begin{pmatrix} K(1) & K(2) \end{pmatrix} = \begin{pmatrix} 6.75 \cdot 10^{-3} & 0 \end{pmatrix} \begin{pmatrix} 0.1985 & 0 \\ 0 & 0.2844 \end{pmatrix}^{-1} = \begin{pmatrix} 0.034 & 0 \end{pmatrix}. \tag{18}$$

Here the control law results in a static feedback of the output $z_1 = h(x) = x_2$. To obtain steady state accuracy we constructed a pre-compensator concerning $x_{2s} = 0.1$:

$$F(x_s) = \Big(C\Big(B(p(x_s))\, K - A(p(x_s)) \Big)^{-1} B(p(x_s)) \Big)^{-1} = K(1) + \frac{\mu_{max}}{k_m + x_{2s} + k_1\, x_{2s}^2} = 2.4\,. \tag{19}$$

The resulting controller (18) and pre-compensator (19) are connected with the bio reactor system (15). The simulated step responses will be shown in the figures below.



Summary

We have shown a new approach to model a basic nonlinear plant as a *linear parameter-varying system* via a nonlinear state transformation. So we have a systematic way to reduce the controller design for input-affine nonlinear plants to LPV-systems. This procedure in combination with a controller design method for LPV-systems was verified on a bio reactor model.

The transformation can extend a *quasi-LPV approach* to achieve a more systematical process. Against conventional controller design via linearization of nonlinear plants locally to equilibrium operating points the transformation in connection with LPV-controller design creates an alternative design approach.

References

1. Becker, G. S., Quadratic Stability and Performance of Linear Parameter Dependent Systems, PhD thesis, University of California at Berkeley, 1993.
2. Bequette, W., Process Dynamics, Prentice Hall, Upper Saddle River, New Jersey, 1998.
3. Crusius, C. A. R. and Trofino, A., Sufficient LMI Conditions for Output Feedback Control Problems, IEEE Transactions on Automatic Control, 44(5), 1999, 1053–1057.
4. Apkarian, P. and Gahinet, P. and Becker, G. S., Self-scheduled H_∞-Control of Linear Parameter-varying Systems: a Design Example, Automatica, 31(9), 1995, 1251–1261.
5. Amato, F. and Garofalo, F. and Glielmo, L. and Pironti, A., Quadratic Stabilization of Uncertain Linear Systems, In: Robust Control via Variable Structure and Lyapunow Techniques, (Eds.: Garofalo, F. and Glielmo, L.), Lecture Notes in Control and Information Sciences, volume 217, Springer-Verlag, London, 1996, 197–211.
6. Huang, Y., Nonlinear Optimal Control: An Enhanced Quasi-LPV Approach, PhD thesis, California Institute of Technology, 1999.
7. Rehm, A. and Allgöwer, F., Self-Scheduled Nonlinear Output Feedback H_∞-Control of a Class of Nonlinear Systems, IfA Technical Report, AUT96-25, ETH-Zürich, 1996.

CONTROLLABILITY VIA AN APPROXIMATION PROBLEM

Stefan Wolfgang Pickl
Darmstadt University of Technology, Germany
Schlossgartenstrasse 7, 64289 Darmstadt
pickl@mathematik.tu-darmstadt.de

Abstract
This paper is concerned with a nonlinear time-discrete dynamical system whose dynamics is described by a system of vector difference equations involving state and control vector functions. It can be seen as a contribution to the investigation of problems of controllability via the solution of an approximation problem. The motivation comes from an actual interdisciplinary research field in the area of environmental systems [4]. The special structure of the developed TEM-model permits two transformations which lead to a solution of the problem of controllability within the smallest number of time-steps, if the problem is solvable [3]. Founded upon these results, the presented algorithm can be determined.

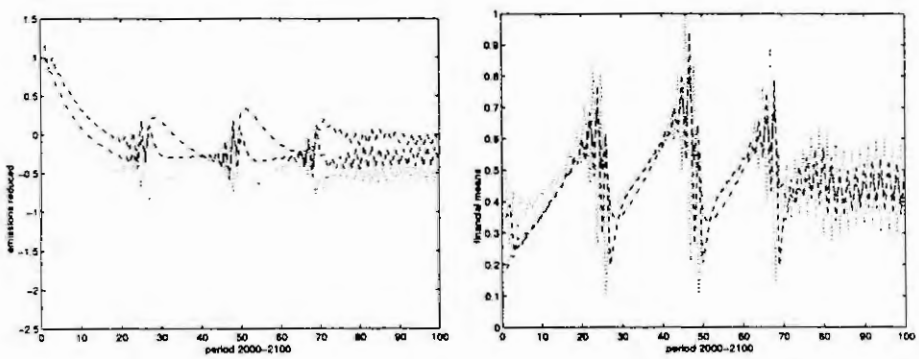
Introduction

The TEM-model (Technology-Emission-Means-model) is a non-linear time-discrete model which describes the economical interaction between several actors. These actors intend to optimize their objective function E_i (reduced emissions which are caused by technologies T_i) by means of expenditures of money or financial means M_i, respectively. The index stands for the i-th player $i = 1, \ldots, n$. The TEM-model was developed to simulate an economic Joint-Implementation Program. Such a process plays a central role in order to fulfil the environment treaties of Rio or Kyoto, respectively. Simulating this economic situation, the actors are linked by technical cooperations and the market. This behavior is expressed by the parameter of the em-matrix which is the central element of the TEM-model:

$$\begin{aligned} E_i(t+1) &= E_i(t) + \sum_{j=1}^{n} em_{ij}(t) M_j(t) \qquad (1) \\ M_i(t+1) &= M_i(t) - \lambda_i M_i(t)[M_i^* - M_i(t)]\{E_i(t) + \varphi_i \Delta E_i(t)\} \end{aligned}$$

E_i	reduced emissions of actor i
M_i	financial means of actor i
λ_i	growth parameter
φ_i	memory parameter

The em_{ij}-parameter describes the effect on the emissions of the i-th actor, if the j-th actor invests money. We can say that it expresses how effective technological cooperations are, which is the kernel of a Joint-Implementation Program. If we let $em_{ij}(t) = em_{ij}^*, t = 0, \ldots, N$, i.e. the economic relationships are constant over a long period, we are able to determine the fixed points of the dynamical system and they are not attractive, even chaotic behaviour is observable:



For a detailed analysis of the TEM-model see [4].

actor	emissions	means	budget	φ	λ	*em*-matrix		
1	-1	0.3	1	11	0.82	1	-0.7	-0.3
2	0.6	0.1	1	11	0.25	-0.8	1	-0.2
3	0.5	0.2	1	11	0.4	-0.9	-0.1	1

data for observing chaos

In order to reach these steady states, an independent institution may influence the trade relations between the actors. Mathematically, the control parameters have to be determined:

The Control Problem

Let us represent the time-discrete dynamical system in (1) by general difference equations added with control vector functions of the form:

$$x_i(t+1) = x_i(t) + f_i(x(t), u(t)) \qquad (2)$$
$$x(t) = (x_1(t), \ldots, x_n(t)) \quad u(t) = (u_1(t), \ldots, u_n(t))$$

for $i = 1, \ldots, n$ and $t \in \mathbb{N}_0 = \mathbb{N} \cup \{0\}$. Here $x_i : \mathbb{N}_0 \to \mathbb{R}^{l_i}$ and $u_i : \mathbb{N}_0 \to \mathbb{R}^{m_i}$ for $i = 1, \ldots, n$ are state and control vector functions, respectively. Furthermore, $f_i : \prod_{j=1}^{n} \mathbb{R}^{l_j} \times \prod_{j=1}^{n} \mathbb{R}^{m_j} \to \mathbb{R}^{l_j}$, $i = 1, \ldots, n$ are given vector functions. In addition we assume, for every $i = 1, \ldots, n$, non-empty sets $X_i \subseteq \mathbb{R}^{l_i}$ and $U_i \subseteq \mathbb{R}^{m_i}$ to be given and require control conditions of the form $u_i(t) \in U_i$ for all $i = 1, \ldots, n$ and $t \in \mathbb{N}_0$ as well as state constraints of the form $x_i(t) \in X_i$ for all $i = 1, \ldots, n$ and $t \in \mathbb{N}_0$. We assume $X_i = \mathbb{R}^{l_i}$ for $i = 1, \ldots, n$ to hold and choose some $N \in \mathbb{N}$.

Then we consider the following **approximation problem:**
Find control functions $u_i : \mathbb{N}_0 \to \mathbb{R}^{m_i}$ with $u_i(t) \in U_i$ for $t = 0, \ldots, N-1$ and $i = 1, \ldots, n$ such that under the conditions

$$x_i(t+1) = x_i(t) + f_i(x(t), u(t)), \quad t = 0, \ldots, N-1 \quad \text{and} \quad x_i(0) = x_{0i}$$

for $i = 1, \ldots, n$ the function value

$$\varphi_N(u) = \sum_{j=1}^{n} (\|x_i(N) - \widehat{x}_i\|_2^2 + \|u_i(N-1)\|_2^2)$$

is as small as possible. If the problem of controllability has a solution, then there is some $N \in \mathbb{N}$ such that for every solution of the above problem it necessarily follows that $u_i(N-1) = \Theta_{m_i}$ and $x_i(N) = \widehat{x}_i$ for $i = 1, \ldots, n$, Θ_{m_i} is the zero vector of $\mathbb{R}^{m_i}$. Hence by solving the above problem, one also obtains a solution of the problem of controllability. The solution of the above problem can be achieved with the help of an algorithm [3]:

The algorithm

We choose control functions $u_i^0 : \mathbb{N}_0 \to \mathbb{R}^{m_i}$ with

$$u_i^0(t) \in U_i \quad \text{for} \quad t = 0, \ldots, N-1 \quad \text{and} \quad i = 1, \ldots, n$$

$$(\text{for instance } u_i^0(t) = \Theta_{m_i} \quad \text{for} \quad t = 0, \ldots, N-1 \quad \text{and} \quad i = 1, \ldots, n)$$

and calculate

$$x_i^0(t+1) = x_i^0(t) + f_i(x^0(t), u^0(t))$$

for $t = 0, \ldots, N-1$ with $x_i^0(0) = x_{0i}$ for $i = 1, \ldots, n$.
Then we construct a sequence

$$(u^k)_{k \in \mathbb{N}_0} \quad \text{in} \quad \left\{ u : \{0, \ldots, N-1\} \to \prod_{j=1}^{n} U_j \right\}$$

and a sequence

$$(x^k)_{k\in\mathbb{N}_0} \quad \text{in} \quad \left\{ x : \{0,\ldots,N\} \to \prod_{j=1}^{n} \mathbb{R}^{l_j} \right\}$$

as follows: If u^k and x^k are given for some $k \in \mathbb{N}_0$, then we determine

$$u_i^{k+1}(t) \in U_i \quad \text{for} \quad t = 0,\ldots,N-1 \quad \text{and} \quad i = 1,\ldots,n$$

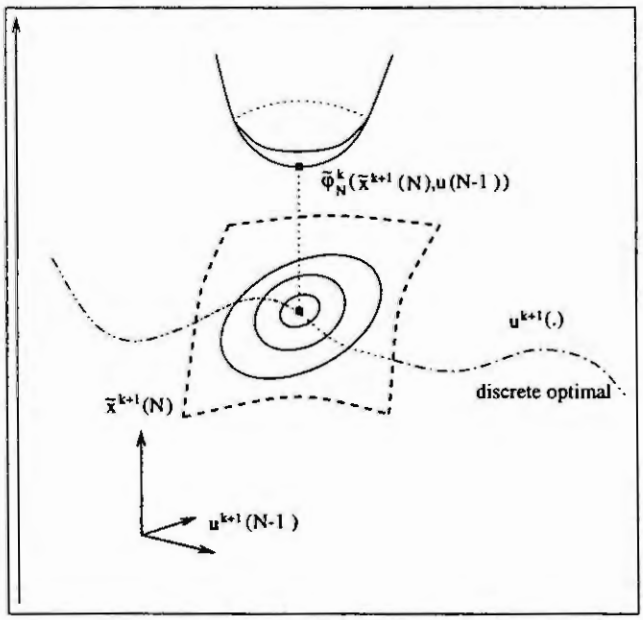
such that under the conditions

$$\tilde{x}^{k+1}(t+1) = \tilde{x}^{k+1}(t) + f(x^k(t), u^{k+1}(t)) \quad \text{for} \quad t = 0,\ldots,N-1$$

and $\tilde{x}^{k+1}(0) = x_0$ the (modified) function value

$$\varphi_N^k(u^{k+1}) \quad = \quad \sum_{i=1}^{n} (\|\tilde{x}_i^{k+1}(N) - \hat{x}_i\|_2^2 + \|u_i^{k+1}(N-1)\|_2^2) \tag{3}$$

becomes minimal. The following figure may reflect this:



Now we obtain the following transformed objective function taking advantage of the special structure of the discrete dynamics which can also be illustrated on the next page.

$$\varphi_N^k(u^{k+1}) \quad = \quad \sum_{i=1}^{n} (\| \sum_{t=0}^{N-1} f_i(x^k(t), u^{k+1}(t)) + x_{0i} - \hat{x}_i\|_2^2 + \|u_i^{k+1}(N-1)\|_2^2)$$

If $u^{k+1}(t)$ has been determined for $t = 0,\ldots,N-1$, then we calculate

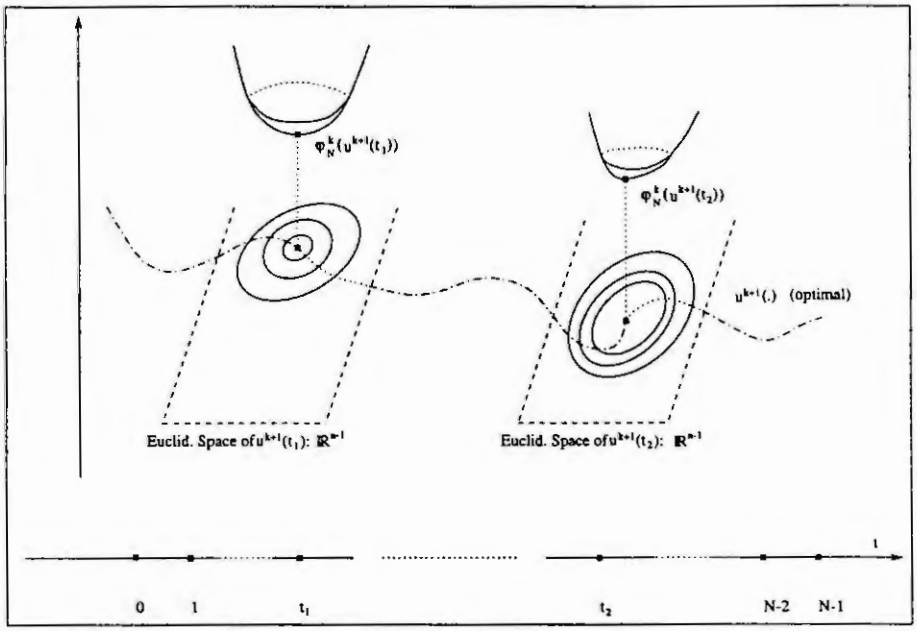
$$x^{k+1}(t+1) = x^{k+1}(t) + f(x^{k+1}(t), u^{k+1}(t))$$

$$\text{for} \quad t = 0,\ldots,N-1 \quad \text{where} \quad x^{k+1}(0) = x_0.$$

If $x^{k+1}(t) = \tilde{x}^{k+1}(t)$ for all $t = 0,\ldots,N$, then we have found a **solution** of the above problem. Otherwise we proceed with u^{k+1} and x^{k+1} instead of u^k and x^k, respectively. Let us make the assumption that all functions

$$f_i : \prod_{j=1}^{n} \mathbb{R}^{l_j} \times \prod_{j=1}^{n} \mathbb{R}^{m_j} \to \mathbb{R}^{l_i}$$

for $i = 1,\ldots,n$ are continuous. Then we have the following



Theorem 1 *If for every* $t \in \{0, \ldots, N-1\}$, *there is some*

$$u(t) \in \prod_{j=1}^{n} U_j \quad \textit{with} \quad u(t) = \lim_{k \to \infty} u^k(t)$$

then $u_i(t)$ *for* $t = 0, \ldots, N-1$ *and* $i = 1, \ldots, n$ *solve the above problem.*

Conclusion

The dynamics of the so called TEM-model decribe an environmental system which contains additionally a technical dimension. In order to reach steady states of the TEM-model which are comparable to the CO_2-values mentioned in the Kyoto protocol the problem of controllability has to be formulated. Via the solution of an approximation problem the problem of controllability can be solved. Taking advantage of the special structure of the discrete dynamics a new algorithm based on a proved existence theorem can be determined.

References

[1] Clarke, F.H., Ledyaev, Yu. S., Stern R.J. and Wolenski, P.R., Nonsmooth Analysis and Control Theory. Springer Verlag, Berlin, Heidelberg, New York 1998

[2] Grützner, R. (Hrsg.), Modellbildung und Simulation im Umweltbereich, ASIM Buchreihe - Fortschritte in der Simulationstechnik, Vieweg Verlag 1997

[3] Krabs, W. and Pickl, S., Controllability of a time-discrete dynamical system via the solution of an approximation problem, Preprint des Fachbereichs Mathematik der TU Darmstadt 1999

[4] Pickl, S., Der τ-value als Kontrollparameter - Modellierung und Analyse eines Joint-Implementation Programmes mithilfe der kooperativen dynamischen Spieltheorie und der diskreten Optimierung, Dissertation am Fachbereich Mathematik der TU Darmstadt (1998), Shaker Verlag (1999)

[5] Nijmeijer, H. and Schaft, A.J. van, Nonlinear Dynamical Control Systems, Springer Verlag, New York, Berlin, Heidelberg 1990

[6] Siciliano, B. and Valavanis, K.P. (Eds.), Control Problems in Robotics and Automation, Lecture notes in control and information sciences: 230, Springer Verlag, London 1998

A NEW ALGORITHM FOR UNKNOWN–INPUT SIMO FIR IDENTIFICATION

U. Soverini, P. Castaldi, R. Diversi and R. Guidorzi
Dipartimento di Elettronica, Informatica e Sistemistica
Università di Bologna
Viale del Risorgimento 2, 40136 Bologna, Italy

Abstract. A new method for estimating the transfer function of single–input multi–output finite impulse response systems driven by an unknown input is proposed. The method is based on the geometric properties of an identification approach originally developed for errors–in–variables models. This technique can be used for systems whose output measurements are affected by additive white noises characterized by different variances. Monte Carlo simulations, performed on models already described in the literature, confirm the effectiveness of the proposed technique.

1 Introduction

Identification problems where the available data can be modeled as outputs of multiple parallel systems driven by an unknown input, arise in many important applications such as data transmission, seismic deconvolution, biomedical data analysis, deblurring of distorted images etc. These problems are usually referred as "blind identification" problems [1]. In most of these applications the unknown system is described by a single–input multi–output (SIMO) finite impulse response (FIR) model with outputs affected by additive noises.

For this problem, two different approaches are usually considered: methods based on second–order statistics of the processes [2], [6], [9]–[11] and approaches based on higher–order statistics [8]. The latter ones are based on optimization technique employing gradient–based algorithms and are characterized by the drawbacks of slow convergence and local minima. Many second–order based methods rely on the so called "cross–relation" property [1], [6], [11]: in a SIMO system, a single output convolved with the impulse response of another FIR equals the output of the first model convolved with the impulse response of the second one. These identification methods are not based on assumptions on the input sequence and yield the true FIR models when the data are noise–free.

It is worth observing that the major part of existing approaches is based on the additional hypothesis that the output noises have equal variances. The identification method proposed in this work relies only on second–order statistics and deals with SIMO FIR systems whose outputs are affected by different unknown amounts of additive white noise. This technique is based on the geometric properties of an identification procedure originally developed for errors–in–variables (EIV) models [3].

The organization of the paper is as follows. Section 2 describes the mathematical setup of the identification problem for unknown–input SIMO FIR models. A solution for this problem is proposed in Section 3 while Section 4 reports the results obtained in the identification of a model already described in the literature. Some short concluding remarks are reported in Section 5.

2 Statement of the problem

Let us consider a linear, discrete, time–invariant, single–input multi–output FIR system, whose noiseless output vector $x(t) = [x_1(t), \ldots, x_M(t)]^T$ is linked to the input $u(t)$ by the transfer function

$$H(z) = \begin{bmatrix} H_1(z) \\ H_2(z) \\ \vdots \\ H_M(z) \end{bmatrix}, \tag{1}$$

where

$$H_i(z) = h_i(0) + h_i(1)\,z^{-1} + \cdots + h_i(L-1)\,z^{-L+1} + h_i(L)\,z^{-L} \qquad (i = 1, 2, \ldots, M) \tag{2}$$

and z is the forward shift operator. The measurements of the processes $x(t)$ and $u(t)$ are not available and we can only measure the noisy output vector

$$y(t) = x(t) + n(t), \tag{3}$$

where $n(t) = [n_1(t), \ldots, n_M(t)]^T$ is the noise vector. With reference to this model, we introduce the following assumption.

Assumption 2.1 – The vector process $n(t)$ is a zero-mean white noise with unknown covariance matrix $D = \text{diag}[\sigma_1^*, \sigma_2^*, \ldots, \sigma_M^*]$ and is uncorrelated with the unknown input $u(t)$.

The problem under investigation is the following.

Problem 2.1 – Starting from the knowledge of the noisy output sequence $y(0), y(1), \ldots, y(N-1)$, estimate the transfer function $H(z)$ and the covariance matrix of the noise D.

In the sequel, it will be shown that Problem 2.1 can be reformulated as a set of single–input single–output (SISO) problems. For this purpose, let us consider a pair of FIR models $H_i(z), H_j(z)$, with $i, j \in \{1, \ldots, M\}$ and $i \neq j$. For the noiseless output signals $x_i(t), x_j(t)$ the following relation holds

$$\begin{aligned} x_i(t) &= H_i(z)\, u(t) \\ x_j(t) &= H_j(z)\, u(t), \end{aligned} \tag{4}$$

i.e.

$$H_j(z)\, x_i(t) = H_i(z)\, x_j(t). \tag{5}$$

Relation (5) shows that the M-dimensional SIMO FIR identification Problem 2.1 can be partitioned into $M(M-1)/2$ SISO problems of the following type.

Problem 2.2 – Given N noisy observations of the outputs $y_i(t), y_j(t)$, determine the noise variances σ_i^*, σ_j^* and the transfer functions $H_i(z), H_j(z)$.

Problem 2.2 has been solved in [5] on the basis of the following considerations. Let us define the coefficient vectors

$$c_m = \Big[\, h_m(L), \ldots, h_m(0) \,\Big]^T \tag{6}$$

and the Hankel matrices

$$X_m(L) = \begin{bmatrix} x_m(0) & \ldots & x_m(L) \\ x_m(1) & \ldots & x_m(L+1) \\ \vdots & & \vdots \\ x_m(N-L-1) & \ldots & x_m(N-1) \end{bmatrix}, \tag{7}$$

with $m = i, j$. On the basis of (5) it is then possible to determine the system parameters by means of the following relation

$$\Big[\, X_i(L) \;\Big|\; X_j(L) \,\Big] \begin{bmatrix} c_j \\ -c_i \end{bmatrix} = 0\,. \tag{8}$$

By introducing the covariance matrix of the bivariate process $[x_i(t) \,|\, x_j(t)]^T$, defined as

$$\hat{\Sigma}_L = \lim_{N \to \infty} \frac{1}{(N-L-1)} \Big[X_i(L) \,|\, X_j(L) \Big]^T \Big[X_i(L) \,|\, X_j(L) \Big], \tag{9}$$

relation (8) leads to

$$\hat{\Sigma}_L \begin{bmatrix} c_j \\ -c_i \end{bmatrix} = 0\,. \tag{10}$$

Remark 2.1 – It is well known [7], [11] that the coefficient vectors $c_m\ (m = i, j)$ can be uniquely determined (up to a scalar factor) by means of (8) or (10), if and only if the transfer functions $H_m(z)\,(m = i, j)$ do not share common zeros and $u(t)$ is persistently exiciting of sufficient order.

3 Identification of unknown–input SIMO FIR systems

In presence of additive noises on the data, only the covariance matrix Σ_L of the noisy bivariate process $[y_i(t) \,|\, y_j(t)]^T$ is available. Under Assumption 2.1 and with $N \to \infty$, the positive definite covariance matrix Σ_L can be decomposed as follows

$$\Sigma_L = \hat{\Sigma}_L + \tilde{\Sigma}_L^*\,, \tag{11}$$

where

$$\tilde{\Sigma}_L^* = \text{diag}\left[\sigma_i^*\, I_{L+1},\, \sigma_j^*\, I_{L+1}\right] \tag{12}$$

is the unknown covariance matrix of the noise process $[n_i(t) \mid n_j(t)]^T$. In this case, the solution of Problem 2.2 can be obtained by analyzing the properties of the sequence of increasing-dimension matrices $(\Sigma_1, \Sigma_2, \cdots)$. These matrices are related to FIR models with different orders $\ell(\ell = 1, 2, \ldots)$ and are constructed on the basis of relations (7) and (9). Let us now consider the family of non-negative definite diagonal matrices $\bar{\Sigma}_\ell = \text{diag}[\sigma_i I_{\ell+1}, \sigma_j I_{\ell+1}]$ such that

$$\hat{\Sigma}_\ell = \Sigma_\ell - \bar{\Sigma}_\ell \geq 0 . \tag{13}$$

Note that every matrix $\bar{\Sigma}_\ell$ can be associated with a point $P_{ij} = (\sigma_i, \sigma_j)$ belonging to the noise plane $\mathcal{R}^{2+}$. These matrices satisfy the following properties [3]:

a) for every ℓ, the set of points associated with matrices $\bar{\Sigma}_\ell$ satisfying relation (13), describes, in $\mathcal{R}^{2+}$, a convex curve whose concavity faces the origin;
b) every point $P_{ij} = (\sigma_i, \sigma_j)$ of this curve is related to the FIR models $c_i(P_{ij}), c_j(P_{ij})$ satisfying the relation

$$\hat{\Sigma}_\ell(P_{ij}) \begin{bmatrix} c_j(P_{ij}) \\ -c_i(P_{ij}) \end{bmatrix} = 0 . \tag{14}$$

where

$$\hat{\Sigma}_\ell(P_{ij}) = \Sigma_\ell - \text{diag}\,[\sigma_i I_{\ell+1}, \sigma_j I_{\ell+1}] \geq 0 ; \tag{15}$$

c) every curve includes all curves associated with higher values of ℓ and the point $P^*_{ij} = (\sigma^*_i, \sigma^*_j)$, corresponding to the actual variances of the noises, belongs to all curves associated with orders $\ell \geq L$;
d) the FIR models corresponding to P^*_{ij} are characterized by the actual coefficients (up to a scalar factor) $c_i(P^*_{ij}), c_j(P^*_{ij})$.

In this theoretical context the search for the solution of Problem 2.2 may thus start from the determination, in the noise plane, of the common point P^*_{ij} [3].

When Assumption 2.1 is not satisfied, and/or the length N of the sequences is finite, the curves corresponding to orders $\ell \geq L$ do not exhibit any common point. Their distance should however decrease in the neighbourhood of P^*_{ij}. This property can be used to obtain an estimate for the order L of the FIR models. Once the order L has been determined, a single solution for the identification problem can be obtained by introducing a suitable model selection criterion [4], [5]. It is thus possible to solve the M-dimensional SIMO FIR identification Problem 2.1 by considering $M(M-1)/2$ scalar problems of type 2.2. This method leads, however, to $M-1$ different estimates of the noise variances and of the associated FIR transfer functions. For this reason, we now introduce a new methodology leading to a congruent solution of the $M(M-1)/2$ scalar problems in the noise space, i.e. to a single estimate for each noise variance σ^*_i $(i = 1, \ldots, M)$.

When Assumption 2.1 is fulfilled and $N \to \infty$, the procedure for the determination of $P^* = (\sigma^*_1, \sigma^*_2, \ldots, \sigma^*_M) \in \mathcal{R}^{M+}$ can be formulated as will be done in the following with reference to a geometric representation that, for the sake of simplicity, will refer to a multivariable problem with $M = 3$. Figure 1 reports the curves of order L related to three possible SISO problems of type 2.2.

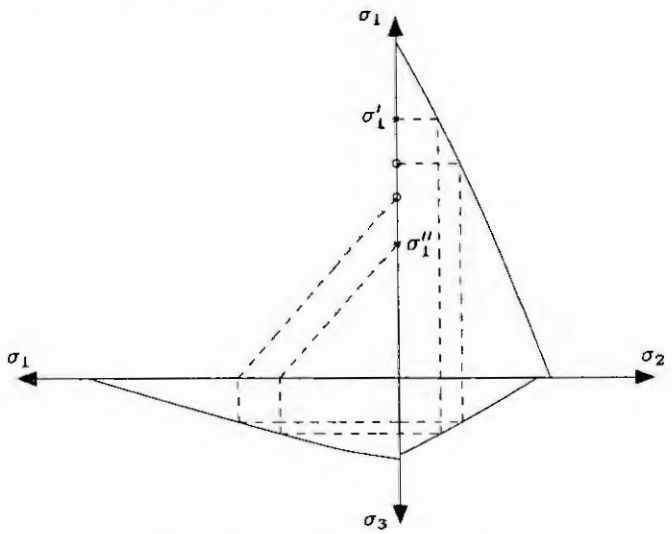


Figure 1: Illustration of the criterion

The convexity of the curves and the fact that each solution of a SISO problem shares a coordinate with the subsequent one, allow the simultaneous treatment of all scalar problems by means of the algorithm described below.

Algorithm 3.1

1) Choose a value σ_1' for σ_1;
2) From σ_1' determine σ_2 so that $P_{12} = (\sigma_1', \sigma_2)$ is a possible solution of the first SISO problem;
3) From σ_2 determine σ_3 so that $P_{23} = (\sigma_2, \sigma_3)$ is a possible solution of the second SISO problem;
4) From σ_3 determine σ_1'' so that $P_{31} = (\sigma_3, \sigma_1'')$ is a possible solution of the third SISO problem;
5) If $\sigma_1'' > \sigma_1'$ consider the interval $S =]\sigma_1', \sigma_1''[$; if $\sigma_1'' < \sigma_1'$ consider the interval $S =]\sigma_1'', \sigma_1'[$. Choose a new value $\sigma_1' \in S$ for σ_1 and go to step 2).
6) Repeat steps 2), 3), 4) and 5) until σ_1 converges, i.e when

$$\frac{|\sigma_1'' - \sigma_1'|}{\sigma_1'} < \epsilon \tag{16}$$

where ϵ is an appropriately small positive number.

Remark 3.1 – Because of the convexity of the curves, when Assumption 2.1 is satisfied and $N \to \infty$, Algorithm 3.1 converges toward the true values $\sigma_1^*, \sigma_2^*, \sigma_3^*$ of the noise variances. Three relations of type (10) can then be used in order to obtain (up to a scalar factor) the actual transfer functions $H_i(z)\,(i = 1, 2, 3)$.

Remark 3.2 – When assumption 2.1 is not satisfied and/or the length N of the sequences is finite, points P_{12}^*, P_{23}^*, P_{31}^* do not belong to the curves; neverthless, thanks to their convexity, Algorithm 3.1 still converges to a single estimate of the noise variances.

Remark 3.3 – Note that the previous algorithm can be easily extended to the case of M outputs and will require the solution of only $M - 1$ SISO EIV problems rather than $M(M-1)/2$.

Remark 3.4 – Note that the congruence of the solution in the noise space is not present in the parameter space. In fact, in correspondence of each scalar problem, a pair of estimates of the transfer functions $H_i(z) \quad (i = 1, \ldots, M)$ are obtained. A single estimate for the FIR models can be achieved by performing a suitable clustering, e.g by means of an average operation.

4 Experimental results

In this section the performance of the proposed approach is illustrated by meaus of a numerical example. A coloured stochastic process of 1000 samples has been used as input for three FIR models, taken from the literature [2] and defined by the following transfer functions

$$H_1(z) = -1.1836 + 0.4906\,z^{-1} - 0.3093\,z^{-2} + 0.4011\,z^{-3} + 0.1269\,z^{-4} - 1.8522\,z^{-5}$$

$$H_2(z) = 1.2965 + 0.0525\,z^{-1} + 0.3410\,z^{-2} - 0.0260\,z^{-3} + 0.3991\,z^{-4} + 0.8817\,z^{-5}$$

$$H_3(z) = 0.9097 - 0.2021\,z^{-1} - 0.4401\,z^{-2} - 1.0153\,z^{-3} - 0.5364\,z^{-4} - 0.0817\,z^{-5}$$

Note that $H_1(z)$ has four unstable roots and $H_3(z)$ has one unstable root. The noiseless output sequences $x_1(t)$, $x_2(t)$ and $x_3(t)$ are characterized by the following standard deviations: $\mathrm{std}\,(x_1) = 25.07$, $\mathrm{std}\,(x_2) = 15.99$ and $\mathrm{std}\,x_3(t) = 11.28$. The output sequences have been corrupted by adding white noises $n_1(t)$, $n_2(t)$ and $n_3(t)$ with standard deviations ranging from 10% to 30% of the standard deviations of the noise–free signals. Note also that the same percent of noise actually corresponds to different amounts of noise on the outputs. The Normalized–Root–Mean–Square–Error (NRMSE)

$$\mathrm{NRMSE} = \frac{1}{\|c_i\|}\sqrt{\frac{1}{R}\sum_{k=1}^{R}\|\hat{c}_i^k - c_i\|^2}, \tag{17}$$

has been employed as a measure of the performance of the proposed method. R is the number of Monte Carlo trials and $\hat{c}_i^k$ is the estimate of the coefficient vector c_i obtained in the k-th trial. Figures 2–4 show the NRMSE versus the percent amount of noise for each pair of model estimates for $R = 100$. These figures show that the

proposed identification method gives a reliable estimate of the model parameters when the amount of noise is not greater than 20%.

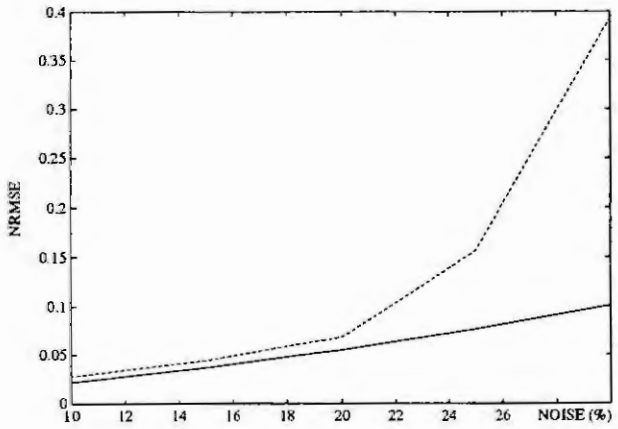


Figure 2: NRMSE associated with the pair of models obtained for $H_1(z)$

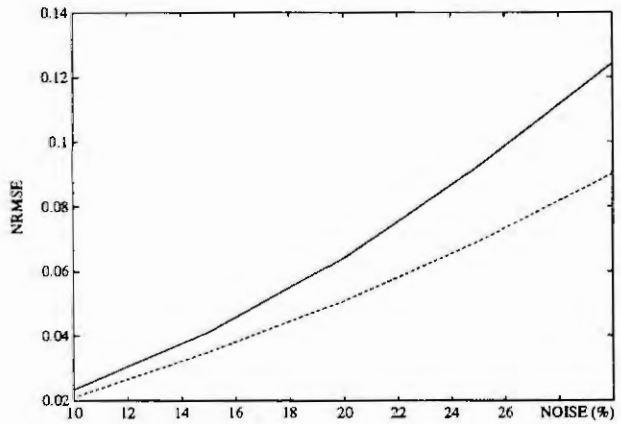


Figure 3: NRMSE associated with the pair of models obtained for $H_2(z)$

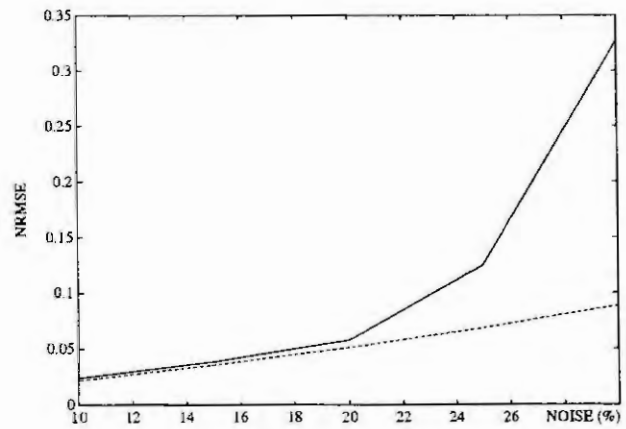


Figure 4: NRMSE associated with the pair of models obtained for $H_3(z)$

Table 1 shows the estimates of the noise variances obtained in the Monte Carlo simulations in five different conditions of signal–to–noise ratio. Good results can be obtained for noise levels up to 30%.

		10%	15%	20%	25%	30%
σ_1^*	true	6.29	14.15	25.15	39.29	56.58
	ident.	6.27 ± 0.23	14.10 ± 0.53	24.99 ± 0.96	38.92 ± 1.60	55.74 ± 2.39
σ_2^*	true	2.56	5.75	10.22	15.97	23.00
	ident.	2.54 ± 0.13	5.71 ± 0.29	10.15 ± 0.51	15.84 ± 0.84	22.84 ± 1.21
σ_3^*	true	1.27	2.86	5.09	7.95	11.45
	ident.	1.27 ± 0.06	2.85 ± 0.14	5.05 ± 0.26	7.86 ± 0.43	11.28 ± 0.62

Table 1: True, estimated values and standard deviations of the noise variances

5 Conclusions

In this work a new blind identification method for single–input multi–output FIR systems has been discussed. The approch is based on the geometric properties of an identification procedure originally developed for errors–in–variables models. The main advantage of this approach, which is based on second-order statistics, consists in its applicability also in environments where the outputs are affected by different amounts of additive noises. The performance of the proposed technique has been illustrated by means of a numerical example taken from the literature. The simulation results show that it gives good estimates of the system parameters for additive noises up to 20%. These results are comparable with those obtained by other approaches which deal with balanced amounts of noise on the channels [2], [11].

References

1. Abed-Meraim, K., Qiu, W. and Hua, Y., Blind system identification. Proceedings of the IEEE, 85 (1997), 1310–1322.

2. Abed-Meraim, K., Moulines, E. and Loubaton, P., Prediction error method for second-order blind identification. IEEE Transactions on Signal Processing, 45 (1997), 694–705.

3. Beghelli, S., Guidorzi, R.P. and Soverini, U. The Frisch scheme in dynamic system identification. Automatica, 26 (1990), 171–176.

4. Beghelli S., Castaldi, P., Guidorzi, R.P. and Soverini U., A robust criterion for model selection in identification from noisy data. In: Proc. of 9th International Conference on Systems Engineering, Las Vegas, 1993, 480–484.

5. Diversi, R., Guidorzi, R.P., Soverini, U. and Castaldi, P., Blind identification of SIMO FIR systems. In: Recent Advances in Signal Processing and Communications, (Ed.: Mastorakis, N.E.) World Scientific Publisher, 1999, 60–64.

6. Gürelli, M.I. and Nikias, C.L., EVAM: an eigenvector–based algorithm for multichannel blind deconvolution of input colored signals. IEEE Transactions on Signal Processing, 43 (1995), 134–149.

7. Hua, Y. and Wax, M., Strict identifiability of multiple FIR channels driven by unknown arbitrary sequence. IEEE Transactions on Signal Processing, 44 (1996), 756–759.

8. Kalouptsidis N., Signal Processing Systems: Theory and Design. John Wiley & Sons, 1997.

9. Moulines, E., Duhamel, P., Cardoso, J.F. and Mayrargue, S., Subspace methods for the blind identification of multichannel FIR filters. IEEE Transactions on Signal Processing, 43 (1995), 516–525.

10. Tong, L., Xu, G. and Kailath, T., Blind identification and equalization based on second-order statistics: a time domain approach. IEEE Transactions on Information Theory, 40 (1994), 340–349.

11. Xu, G., Liu, H., Tong, L. and Kailath, T., A least–squares approach to blind channel identification. IEEE Transactions on Signal Processing, 43 (1995), 2982–2993.

FILTER-CHAIN MODELS FOR IDENTIFICATION OF NONLINEAR DYNAMICAL SYSTEMS

K. Voigtländer and H.-H. Wilfert
Fraunhofer-Institute for Information and Data Processing (IITB)
Zeunerstr. 38, D-01069 Dresden
e-mail: voigtlae@ivi.iitb.fhg.de

Abstract. The modeling of nonlinear dynamical systems is considered in this paper. It is assumed that only sampled input/output data of the system under investigation are available. The most popular way for creating a black-box-model is a common nonlinear difference-equation-approach. Some basic features of such an approach are related to the corresponding properties of a filter-chain model consisting of a linear dynamical system followed by a nonlinear readout map. The filter-chain model has some very good structural characteristics but needs to be optimized with respect to its approximation efficiency. So the construction of a suitable filter system - which enables an efficient modeling - and the construction of an adjusted nonlinear readout map from a given data set is considered.
To illustrate the relation between a proper filter selection and an efficient modeling some theoretical reflections concerning an optimal filter design for an approximation of a given nonlinear system are presented afterwards. The both discussed methods are based on a VOLTERRA-Kernel representation and a state space description of the plant and yields to an adjusted filter-chain model and a bilinear filter model respectively.

1. The nonlinear difference-equation-approach and the filter-chain model

Within this section both models are established and compared with respect to some basic features arising from its structural properties.
Concerning the assumed time-continuous character of the process it is fair to ask for an appropriate time-continuous model. To deal with the acquired sampled input-output data a corresponding time-discrete version of the continuous model is necessary. In general it is impossible to obtain an analytic discrete version of a time-continuous differential model and so a quasi continuous simulation (with a numerical integration) is necessary. To avoid such a time consuming approximation it is common practice to cancel the request for a time-continuous model and to establish a time-discrete difference-equation model structure a priori (Fig. 1). Unfortunately, once the parameterization is done the model cannot be re-mapped into a continuous form.

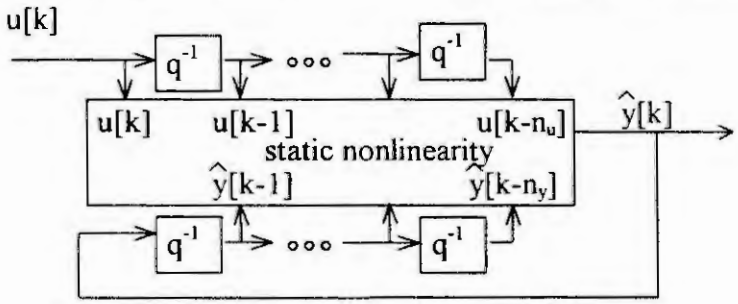


Fig. 1 Nonlinear difference-equation model

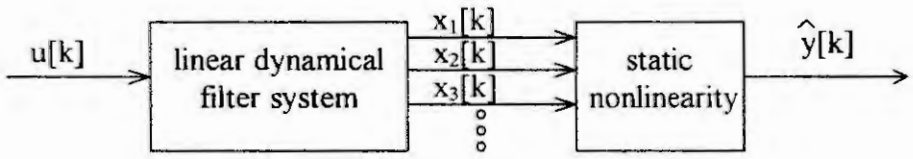


Fig. 2 General form of the nonlinear filter-chain-model

On the contrary the time continuous filter-chain model - its general form consists of a linear dynamical system followed by a nonlinear readout map - can be exactly mapped into a time discrete analytic equivalent by

assuming a hold element for the interpolation of the input signal between two samples. So the continuous model can be parametrized by using the fitting properties of the time-discrete equivalent (Fig. 2) .
Considering stability properties, aspects of parameter estimation and occasions for a structure selection within the static nonlinearity one has to admit that the filter-chain model offers excellent features (Table 1). Especially the parameterization of the difference equation model with respect to an equation error is critical because - in contrast to linear theory - there is only a mysterious link between equation and output error. Even the output error can become infinite (caused by an instable model) while the equation error is sufficient small [5].

Table 1 Comparison of difference-equation model and filter-chain model

feature	difference-equation model	filter-chain model
relation between time continuous and time discrete model	no analytic discrete equivalent existing; numerical integration with respect to a selected interpolation element necessary	analytic calculation with respect to a selected interpolation element possible
guarantee of global asymptotic stability	an a posteriori proof generally impossible; an a priori guarantee yields to unfeasible approximation restrictions	stability of the linear dynamic part sufficient and easy to guarantee
parameterization of the nonlinearity	feasible with respect to an insufficient equation error; very difficult with respect to the output error	feasible with respect to the output error
structure selection	only possible with respect to an insufficient equation error	possible with respect to the output error
considered process class	nonlinear state space description with a unique and always defined solution	fading memory systems [1]: a unique stationary solution has to be reached asymptotically for any bounded input

While a lot of structural advances of the filter-chain model have been recognized, where is the drawback of this approach? The answer can be found very easily by checking some aspects of approximation efficiency. In the case of the difference-equation model the number of delay elements determining the input dimension of the difference model is strongly related to the order of the process considered. It has to be increased only if the output-map has not a unique inverse map. Unfortunately in the case of the filter-chain model the number of filters and so the input dimension of the readout-map must generally tend to infinity for an exact representation. But with an appropriate chosen filter system the input dimension can remain small while the accuracy of the approximation is sufficient. So the construction of a suited filter-system is a milestone within the estimation of a filter-chain model.

2. Identification of filter-chain models

The identification process involves the suitable determination of the eigenvalues of the filter and the construction of an adjusted readout map.

The construction of a suited filter-system from measured input/output data
The most popular example of a simple not adjusted filter system (a simple tapped delay line) is involved in VOLTERRA's famous approach. Consequently a large number of delay-filters is required for a reasonably well approximation and yields to a huge number of parameter to estimate within the polynomial readout map. One can ameliorate this problem by replacing the tapped delay filters with other filters. If these filters are well adjusted a reasonable reduction of the required number of states is possible. WIENERS's model with LAGUERRE filters or the use of KAUTZ filters [9] are examples of such well known filter systems. Recent papers discussing the problem of a suited filter parameterization for an efficient approximation of linear system [6] recommend adjustments based on the impulse response of the plant. For nonlinear systems only rules of thumb concerning the transition time of the nonlinear plant are available [4]. Obviously such recommendations are only helpful when the plant has dominant linear parts and weak nonlinearities. To deal with the crucial problem of a suitable filter parametrization in a more general way a geometrical approach is suggested now. One possible approach for a filter-design based on measured process-data can be derived according to geometrical reflections. The filter-operators of the model perform an embedding [7] of the input series into the state-space of the model. The followed readout map provides a static mapping of this state-space to the model

output. To make the model output equal to the plant output one has to establish such an embedding which allows a static mapping from state-space of the model to the plant output (apart from some measurement disturbances). Hence a plot of the plant output versus the state-space of the model is useful to judge the embedding quality because it has to prove a static dependence of the data. Fig. 3 shows embeddings with a first-order low-pass-filter. While a low and a large time constant result in a very poor embedding a proper chosen time constant yields to an appropriate arrangement of the data. For a further separation an embedding with two filters is necessary. Similar the parameters of these filters have to be tuned in such a way that the embedded data form a surface as flat as possible.

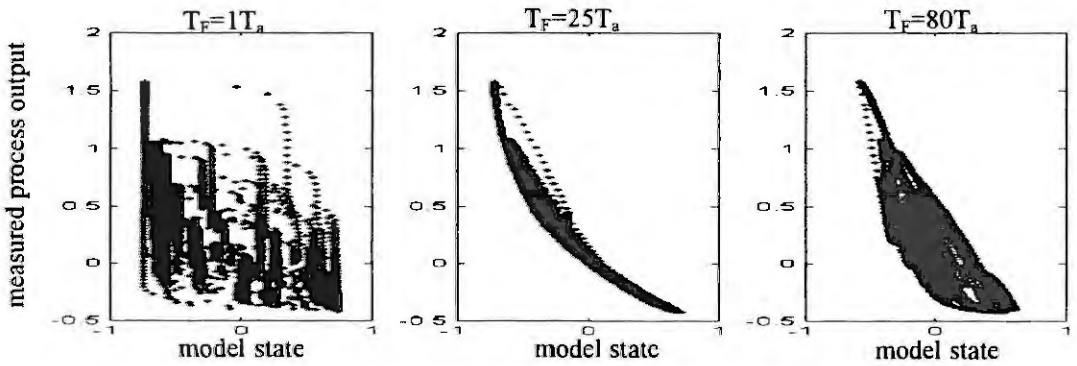


Fig. 3 Measured process output over filter-embedded input series

Summary, a model state-space must be created which yields to a unique assignment of the embedded input data to the measured process output. The optimal model filters can be found by an optimization of the filter parameters with respect to the „roughness" of embedded data. Two proposals for the measurement of this „roughness" are shown in Fig. 4. While the box-counting method provides a cumulative estimation of local variances a parametric measurement calculates the residuum of the embedded data with respect to a smooth parametric approach.

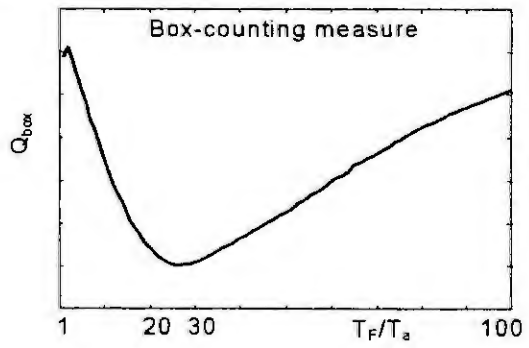


Fig. 4a
Cumulative local (100 boxes) variance

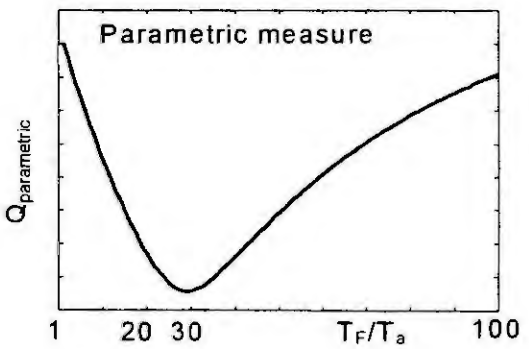


Fig. 4b
Residuum of a 3rd order polynomial approach

In praxis the demonstrated optimization might be limited to about two eigenvalues. A further suited embedding can be achieved by a repeated use of the optimized eigenvalues in the sense of a filter-chain.

The construction of an adjusted readout map

After a suited filter system has been chosen or optimized an adjusted model output (readout map) must be established. A simple but universal approach for an output map is given by

$$\hat{y}[k] = \sum_i \hat{p}_i \varphi_i\left(x[k]\right)$$

were $\{\varphi_i\}$ is a specific approximation basis. Personal preferences are crucial for the actual choice of a basis (polynoms, radial basis functions, sigmoidal functions, wavelets, splines, walsh functions, ...) and further discussion is superfluous without any a priori information of the function to be approximated.

Although the choose of a suited filter system reduces the number of required states enormously (e.g. in comparison with the classical VOLTERRA model) any nonlinear expansion will still yield to a large number of p-parameters for an accurate approximation. Unfortunately the estimated number of parameters must be limited

for a good regression because of output disturbances. So an optimal collection of regressors finally to represent the model has to be estimated using structure selection methods [3]. For the stepwise choice of most important regressors from the basis-pool a forward regression algorithm [2] has been proved to be a suitable one. The algorithm is useful to establish a ranked orthogonal regressor sequence. Finally the optimal number of orthogonal regressor which has to be incorporated in the model must be determined. The recommended way of cross-validation is a special generalization of PRESS-algorithm [2] where parts of the data are sequential excluded from the parametrization process. The resulting parameter-sets will be applied on the removed data part respectively. The partitioning of the data has to be done with respect to the excitation of the process and includes disturbances. By monitoring this cross-validation error while the forward regression is in progress an optimal model complexity can be found. So it is possible to establish an adjusted readout map with respect to special signal considerations (measurement time, excitation, disturbances).

3. Derivation of suitable filter systems for the approximation of nonlinear systems

The previous given approaches for a filter design are based on measured input/output and are therefore suited for an identification process. But for a good understanding of the relation between a proper filter parametrization and an efficient modeling some theoretical reflections concerning an optimal filter design for an effective approximation of a given nonlinear system are very useful.
Within this chapter two different approaches for the construction of an optimal filter system are presented. The first method is based on a VOLTERRA kernel representation of the plant and results in a proper adjustment of a first order filter for the approximation of the plant with a filter-chain model. The second calculation yields to a bilinear filter system which eigenvalues are related to the eigenvalues of the linearized plant at the equilibrium point.

Kernel approach
Starting with the LAPLACE representation of the VOLTERRA kernels of the plant

$$K^1(s_1)\ ;\ K^2(s_1,s_2)\ ;\ K^3(s_1,s_2,s_3)\ ;\ \ldots$$

one can replace the complex variables s with a filter operator F of a considered first order filter F(s). An expansion of the resulting kernels as a TAYLOR series yields to

$$K^1\big(F(s_1)\big)=\sum_{i=0}^{\infty} c_i F^i(s_1)\ \ ;\ \ K^2\big(F(s_1),F(s_2)\big)=\sum_{i=0}^{\infty}\sum_{j=i}^{\infty} c_{ij}F^i(s_1)F^j(s_2)\ \ ;\ \ \ldots$$

which have a simple filter-chain representation (Fig. 5) [8]. Further the coefficients c_i, c_{ij}, ... represent the TAYLOR expansion of the required readout-map to be established.

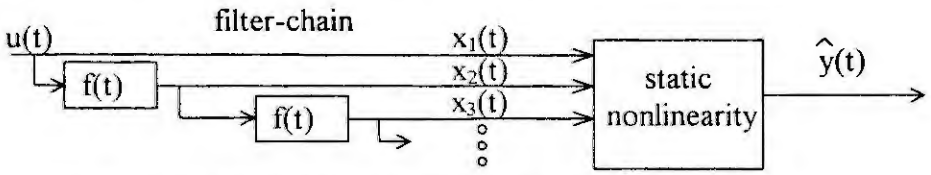


Fig. 5 Special form of the nonlinear filter-chain-model

Mainly the c-coefficients depend on the parameters of the applied filter F resp. its impulse response f(t). Hence the goal is a filter parametrization which yields to a fast decrease of the c's towards zeros. This ensures a feasible approximation quality even with only a few filters involved in the final model. Even though an analytic estimation of an adjusted filter-set requires knowledge about the VOLTERRA kernels of the system to be modeled this method yields to a good understanding of the relation between filter characteristics and approximation efficiency.

CARLEMAN approach
The second approach is based on the state-space description of the analytic linear system

$$T\,\underline{\dot{x}} = \underline{a}(\underline{x}) + \underline{b}(\underline{x})u(t)$$

$$y(t) = \underline{c}^T\,\underline{x}$$

The introduction of a new state vector $\underline{x}^{\otimes}$ which contains products of the original state up to a specific order yields to a bilinear description

$$T\underline{\dot{x}}^{\otimes} = A\underline{x}^{\otimes} + N\underline{x}^{\otimes}u(t) + Bu(t)$$

$$y(t) = (\underline{c}^{T} \quad 0 \quad \ldots)\underline{x}^{\otimes}$$

of the above system. From this representation it is now possible to derive a kernel representation [8] which can be realized as a junction of filter systems S_i with a linear output map (Fig. 6). The eigenvalues Λ_1 of the systems S_1 must be equal to the eigenvalues of the linearized plant at the equilibrium point. The system S_2 consists of filters equal to those of system S_1 and contains filters whose eigenvalues Λ_2 can be calculated by summing the eigenvalues of S_1 in pairs. The additional eigenvalues Λ_3 within the system S_3 result in a further pair-wise summation of the eigenvalues Λ_1 and Λ_2.

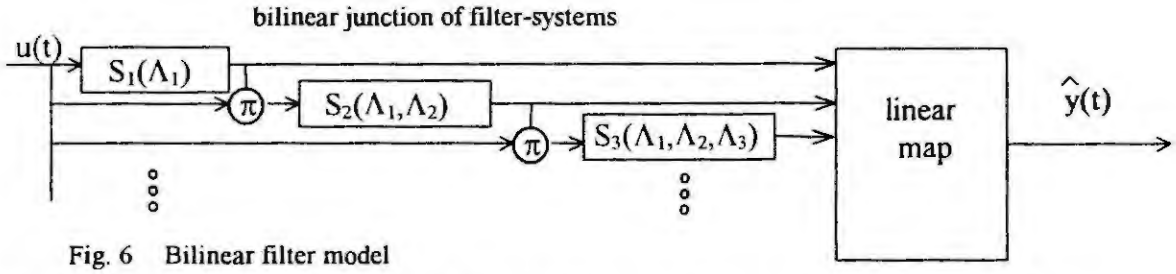


Fig. 6 Bilinear filter model

While the merely linear output map is an advantage of this model the number of state signals inflationary increases from one of filter system to the next. This is because every input signal of a filter-system has to pass each filter-element. Compared with the simple calculation of a time discrete equivalent of the filter-chain model (Fig. 5) it is still possible but time consuming to get a time discrete equivalent of the bilinear model for the use with sampled data since an exponential matrix must be calculated within every time step. But at least with this approach an exact representation of the first N kernels is possible with N filter-systems parameterized in relation to the eigenvalues of the linearized plant.

Summary

The superior structural properties of the filter-chain model have been worked out. The analysis of the approximation task yields to suggestions for a suitable filter parametrization. Considering the importance of a proper filter parametrization for a high approximation efficiency two different approaches are given for the identification of a filter-chain model from sampled input-output data.

References

[1] Boyd, S.; Chua, L.O.: Fading memory and the problem of approximating nonlinear operators with volterra series. IEEE Trans. on CAS, vol. 32 (1985), no. 11, pp. 1150-1161.

[2] Draper, N.R.; Smith, H.: Applied regression analysis, John Wiley, New York 1981.

[3] Haber, R.; Unbehauen, H.: Structure identification of nonlinear dynamic systems - a survey on input/output approaches. Automatica, vol. 26, no. 4 (1990), pp. 651-677.

[4] Kurth, J.: Identifikation nichtlinearer Systeme mit komprimierten Volterra-Reihen. Dissertation RWTH-Aachen, Fortschritt-Berichte VDI, VDI-Verlag, Düsseldorf 1995. Reihe 8, Nr. 459.

[5] Nelles, O.: On the identification with neural networks as series-parallel and parallel models. International Conference on Artificial Neural Networks (ICANN), Paris, October 1995.

[6] Oliviera a Silva, T.: On the determination of optimal pole position of Laguerre filters. IEEE Trans. Signal Process., vol. 43, no. 9, pp. 2079-2087.

[7] Packerd, N.H.; Crutchfield, J.P.; Farmer, J.D.; Shaw, R.S.: Geometry from a time series. Physical review letters, vol. 45 (1980), no. 9, pp. 712-716.

[8] Rugh, W.J.: Nonlinear system theory. The Johns Hopkins University Press, Baltimore and London 1981.

[9] Wahlberg, B.; Mäkilä, P.M.: On approximation of stable linear dynamical systems using Laguerre and Kautz functions. Automatica, vol. 32 (1996), no. 5, pp. 693-708.

MODELLING, IDENTIFICATION, AND SIMULATION OF A HYDROSTATIC TRANSMISSION

G. Hametner
Department of Mechanical Engineering, Technical University Vienna
Wiedner Hauptstraße 8–10, A–1040 Wien
Tel: *43-1-58801-30311, e-mail: Gudrun.Hametner@tuwien.ac.at

Abstract. For the identification of dynamic systems from measured data, various standard procedures are available. In engineering applications, these procedures may have to be adapted and extended for several reasons. Apart from the measured data, there may be some additional knowledge about the system that should be incorporated into the identification. If there is a large amount of measured data, it must be decided which part of the data shall be used for the identification; it can be expected that the identification result improves as the amouut of data is increased. If there is noise in the measured signals, they may be filtered before the identification is carried out. For a hydrostatic transmission system, all these problems have been dealt with including the identification of time-varying load parameters. The quality of the identification result is proved by a comparison of measured signals and simulation results.

1 Introduction

A common approach to system identification based on measured data is to estimate the coefficients of a difference equation; apart from the system order, no further information about the system is utilized [3].

The dynamic behaviour of hydrostatic systems can be modelled by ordinary differential equations. These contain some parameters that can directly be measured, like masses, cylinder cross-sections, etc. Knowing the structure of an ordinary differential equation and even some of its coefficients means having additional information apart from the system order. Chen [2] does not make use of this fact, resulting in multiple solutions when he calculates the differential equation coefficients of a hydrostatic positioning drive from identified difference equation coefficients. Pollmeier et al. [6] present an identification method for condition monitoring in hydrostatic systems, which is based directly on the system differential equations.

In the present paper, the unknown model parameters of a hydrostatic transmission are to be identified from measured data. The measured signals are rather noisy, which makes it difficult to find reliable values for the unknown model parameters. It is therefore essential to use all the information available about the system. This results in an approach similar to Pollmeier et al. [6]; moreover, various ways of data and model validation are presented.

2 Hydrostatic transmission model

As opposed to a hydrodynamic drive, where a pump drives a turbine, a hydrostatic transmission consists of a pump and a hydraulic motor [1]; the motor is driven by the potential energy of the oil pressurized by the pump. Figure 1 shows a simplified hydraulic circuit diagram of the hydrostatic transmission under consideration.

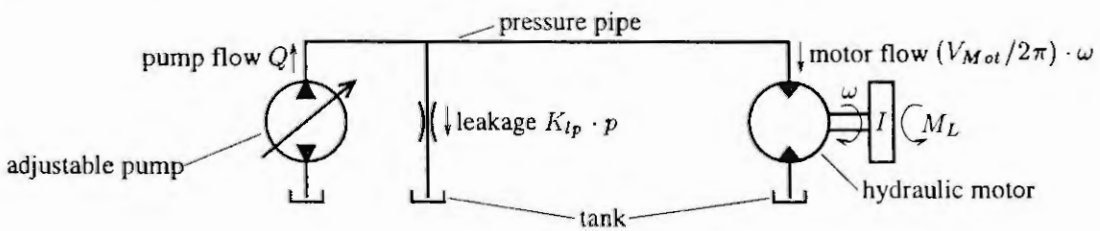


Figure 1: Simplified circuit diagram of hydrostatic transmission.

The pump produces an adjustable oil flow Q. The leakage is modelled by the product of a constant leakage coefficient K_{lp} and the pressure p in the pressure pipe. The oil flow passing through the hydraulic motor is equal to $(V_{Mot}/2\pi) \cdot \omega$, where ω denotes the angular speed of the motor shaft and V_{Mot} is the

oil volume passing the motor during one revolution of the shaft. With the hydraulic capacitance C_H of the pressure pipe, the continuity equation reads

$$C_H \dot{p} = Q - K_{lp} p - \frac{V_{Mot}}{2\pi} \omega, \tag{1}$$

where dp/dt is denoted as $\dot{p}$. The equation of motion has the form

$$I\dot{\omega} + d\omega = \frac{V_{Mot}}{2\pi} p - M_L, \tag{2}$$

where I is the mass moment of inertia of the machine driven by the motor, d is the damping coefficient, $(V_{Mot}/2\pi) \cdot p$ is the torque transmitted through the motor shaft, and M_L is the torque exerted on the machine by external loads. The measurement of the angular speed ω can be approximated by a first-order delay, such that the relation between ω and the measured value $\bar{\omega}$ is described by

$$T\dot{\bar{\omega}} + \bar{\omega} = \omega. \tag{3}$$

Equations (1), (2), and (3) contain two well-known coefficients, I and $V_{Mot}/2\pi$. The coefficients C_H, K_{lp} and T are constant with unknown values; they shall therefore be called the unknown constant model parameters. The coefficient d and the torque M_L shall be denoted as time-varying load parameters since they depend on time-varying external loads.

3 Experimental setup and measurements

In the practical application considered, the hydrostatic transmission operates in closed loop, and the angular speed $\bar{\omega}$ is controlled. Using this configuration, experiments have been carried out and the quantities Q, p, and $\bar{\omega}$ have been measured. A MATLAB/SIMULINK block diagram of the experimental setup is shown in Fig. 2. In the block diagram, the unknown values and functions have been replaced by question marks; the signals that have been measured during the experiments are indicated by scope symbols.

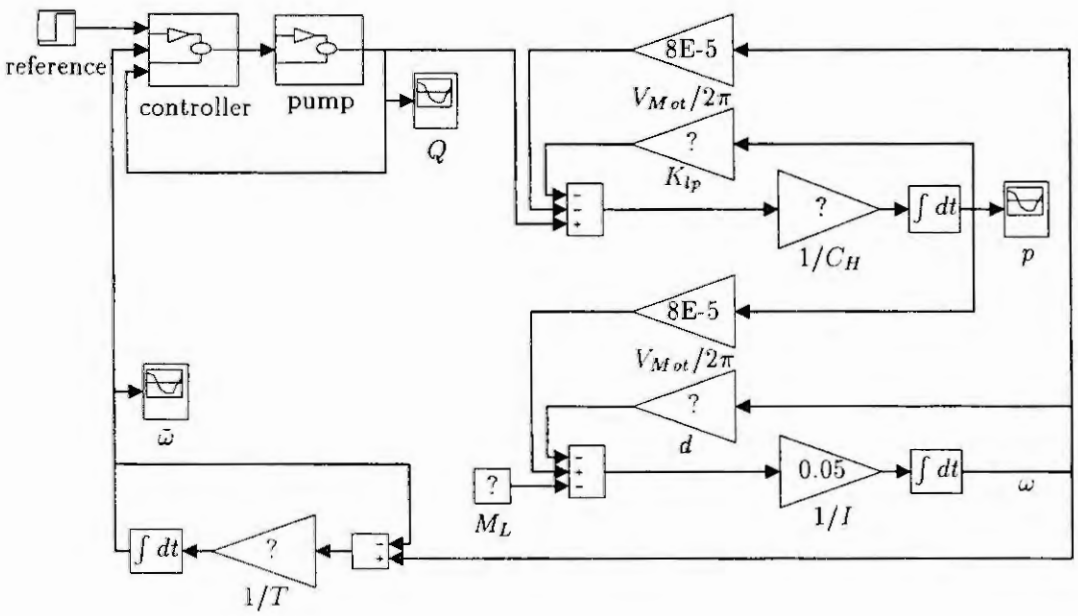


Figure 2: Block diagram of hydrostatic transmission in closed loop.

4 Identification of unknown constant model parameters

For the identification of the parameters C_H, K_{lp}, and T, Eqn. (3) is inserted into Eqn. (1). Q, p, $\dot{p}$, $\bar{\omega}$, and $\dot{\bar{\omega}}$ are substituted by discrete-time values according to the bilinear Tustin-approximation [4]. This

yields the difference equation

$$\mathbf{x}^T(k)\,\boldsymbol{\theta} = y(k), \tag{4}$$

where

$$\mathbf{x}^T(k) = \left[\frac{p_k - p_{k-1}}{T_S} \quad \frac{p_k + p_{k-1}}{2} \quad \frac{V_{Mot}}{2\pi} \cdot \frac{\bar{\omega}_k - \bar{\omega}_{k-1}}{T_S}\right] \tag{5}$$

with the time step $T_S = 0.4$ ms and $p_k = p(kT_S)$, the parameter vector

$$\boldsymbol{\theta} = [C_H \quad K_{lp} \quad T]^T, \tag{6}$$

and

$$y(k) = \frac{Q_k + Q_{k-1}}{2} - \frac{V_{Mot}}{2\pi} \cdot \frac{\bar{\omega}_k + \bar{\omega}_{k-1}}{2}. \tag{7}$$

Equation (4) can be applied for $k = 1, 2, \ldots, n$, which can be written in matrix notation as

$$\mathbf{X}(n)\boldsymbol{\theta} = \mathbf{y}(n). \tag{8}$$

Using the least squares method [3], the estimate $\hat{\boldsymbol{\theta}}$ of the parameter vector is given by

$$\hat{\boldsymbol{\theta}}(n) = (\mathbf{X}^T(n)\mathbf{X}(n))^{-1}\mathbf{X}^T(n)\mathbf{y}(n). \tag{9}$$

A way of validating the estimate is to check the convergence of the results for $n = N, N+1, \ldots, N+M$. If $\hat{\boldsymbol{\theta}}(n)$ were calculated from Eqn.(9) each time, M+1 matrix inversions would have to be performed. To save computation time, only the initial estimate $\hat{\boldsymbol{\theta}}(N)$ is calculated from Eqn.(9). For $\hat{\boldsymbol{\theta}}(N+1), \ldots, \hat{\boldsymbol{\theta}}(N+M)$, the recursive least squares algorithm is used, which is given by the following equations [5]:

$$\boldsymbol{\gamma}(n) = \frac{1}{1 + \mathbf{x}^T(n+1)\mathbf{P}(n)\mathbf{x}(n+1)}\mathbf{P}(n)\mathbf{x}(n+1), \tag{10}$$

$$\hat{\boldsymbol{\theta}}(n+1) = \hat{\boldsymbol{\theta}}(n) + \boldsymbol{\gamma}(n)\left(y(n+1) - \mathbf{x}^T(n+1)\hat{\boldsymbol{\theta}}(n)\right), \tag{11}$$

$$\mathbf{P}(n+1) = \left(\mathbf{I} - \boldsymbol{\gamma}(n)\mathbf{x}^T(n+1)\right)\mathbf{P}(n); \tag{12}$$

the matrix $\mathbf{P}(N)$ is given by

$$\mathbf{P}(N) = \left(\mathbf{X}^T(N)\mathbf{X}(N)\right)^{-1}. \tag{13}$$

When carrying out Eqns. (10), (11), and (12), various sets of measured data from different experiments are available. One of these data sets has to be selected for the calculation of the final estimation result. This can be done by the convergence check as mentioned above; another indication for suitable data is the presence of significant vibrational amplitudes, i.e. there is no problem to identify the coefficients of the signals' derivatives. To reduce the influence of noise, all signals are filtered by using a first-order lowpass filter with a time constant of 100 ms before the identification is carried out. Figure 3 shows the convergence of the estimated values of C_H, K_{lp}, and T from the most suitable set of measured data.

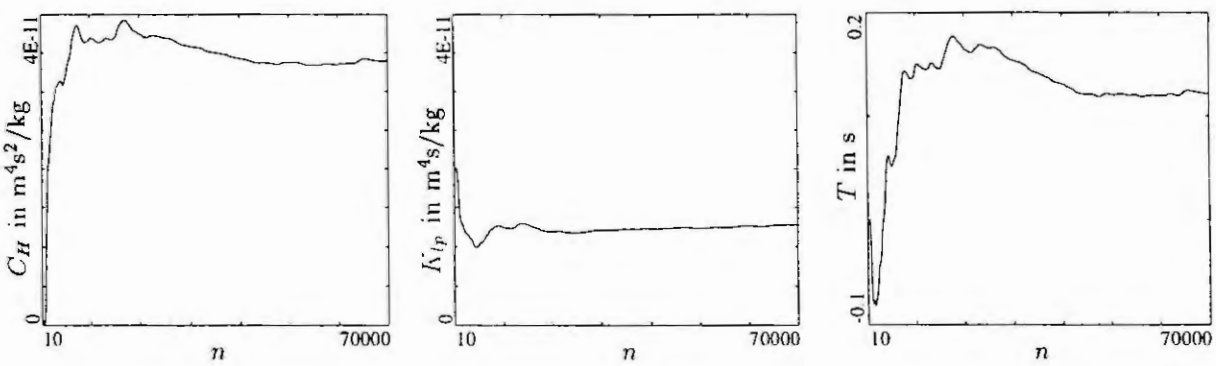


Figure 3: Estimation results for $n = 10, 11, \ldots, 70000$.

5 Identification of time-varying load parameters

Using Eqns. (2) and (3) and the values of C_H, K_{lp}, and T as identified in section 4, the load parameters d and M_L may be identified by a similar procedure. However, in this way only the mean values of d and M_L can be obtained. To estimate the functions $d(t)$ and $M_L(t)$, a forgetting factor $\lambda = 0.99$ is introduced into the recursive least squares equations according to [3]. The measured signals must not be filtered in this case.

6 Simulation of the closed-loop system

Having determined the unknown constant model parameters and the time-varying load parameters, the closed-loop system in Fig. 2 can be simulated. The simulation results for Q, p, and $\bar{\omega}$ should match the measured signals; this fact has been used for model validation and for the choice of the forgetting factor λ. In Figs. 4 and 5, measured signals and simulation results are compared for two different experiments.

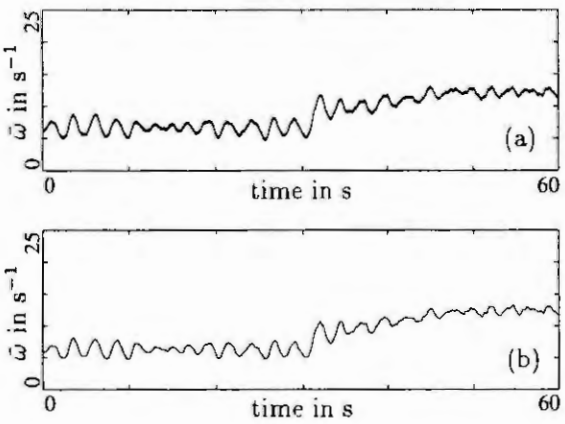


Figure 4: Comparison between measured (a) and simulated (b) angular motor speed $\bar{\omega}$, experiment near the stability limit.

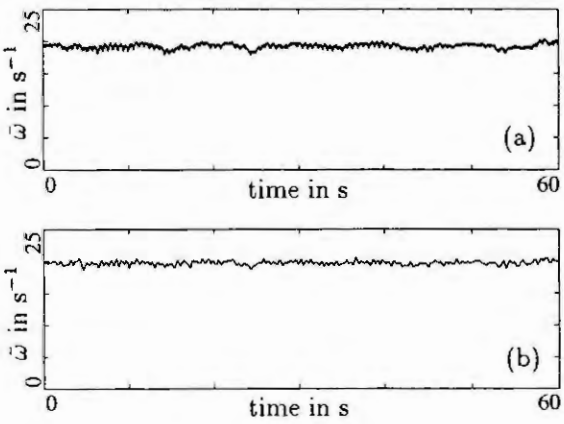


Figure 5: Comparison between measured (a) and simulated (b) angular motor speed $\bar{\omega}$, forced vibrations experiment.

7 Conclusions

There is a good agreement between the simulation results and the measured signals, even though the identification was based on rather noisy data. The model is now ready to be used for the design of a state controller and for the simulation of the closed-loop system under various operating conditions.

References

1. Backé, W., Servohydraulik. 6. Auflage, Institut für hydraulische und pneumatische Antriebe und Steuerungen der RWTH Aachen, 1992.

2. Chen, M.-F., Zustandsgrößengenerierung und ihre Rückführung bei fluidtechnischen Positionierantrieben. Dissertation, RWTH Aachen, 1995.

3. Isermann, R., Lachmann, K.-H., Matko, D., Adaptive Control Systems. Prentice Hall International (UK) Ltd, 1992.

4. Jörgl, H. P., Repetitorium Regelungstechnik, Band 2, Oldenbourg, Wien, 1994.

5. Phillips, C. L., and Nagle, H. T., Digital Control System Analysis and Design. Third Edition, Prentice Hall, Englewood Cliffs, New Jersey, 1995.

6. Pollmeier, K., Größenreuther, F., Burrows, C. R., Edge, K. A., Identification of nonlinear components for condition monitoring in hydraulic systems. In: Proc. 12. Aachener fluidtechnisches Kolloquium, Aachen, 1996, Band 2, 239 - 256.

CANONICAL VARIATE ANALYSIS NON-LINEAR STATE SPACE MODELLING

A. Simoglou, E.B. Martin and A.J. Morris
Foresight Centre for Process Analytics and Control Technology
University of Newcastle upon Tyne, Newcastle upon Tyne, NE1 7RU, U.K
e.b.martin@ncl.ac.uk

Abstract. The issue of modelling non-linear systems using Hammerstein and Wiener type models is investigated in this paper. Both models involve the computation of a linear dynamic part and a static non-linear element. The multivariate subspace projection technique of Canonical Variate Analysis (CVA) is used to identify the parameters of the state space model which is used to approximate the linear dynamic element, whilst the non-linear static element is described in terms of polynomial functions. Initially, a published CVA-Hammerstein model is reviewed. The methodology is improved by proposing a more efficient way to calculate the orders of the system. In addition a novel Wiener model is proposed based on a CVA state space model representation. The two model types are compared using a benchmark pH neutralisation process under various levels of noise. The proposed CVA-Wiener model gives improved modelling performance compared to the equivalent Hammerstein model in terms of providing predictions that are more accurate.

1. Introduction

Non-linear behaviour becomes much more common as processes are operated closer to their operational constraints. Describing non-linear systems using empirical models is a difficult and challenging task. This is due to the complexity of these systems and the fact that in many situations there is no *a priori* knowledge of the nature of the system dynamics and the form of the non-linearity. Many model forms and solution algorithms have been proposed for the identification of non-linear systems. This paper focuses on the development of block oriented models. This family of models is popular because of their structural simplicity and the fact that they are representative of a wide range of non-linear systems. Block oriented models are described in terms of a series of linear dynamic and non-linear static blocks. Where the linear dynamic block follows the static non-linear block, the model is termed a Hammerstein model. If the static non-linear block follows the dynamic block then the representation is known as a Wiener model. In the literature, the non-linear static block has typically been approximated by a polynomial function, although other types of non-linear functions can be used, e.g. neural networks. The linear dynamic element may take the form of any linear dynamic model such as a transfer function, a state space model or an autoregressive with exogenous inputs (ARX) model. Traditionally, non-linear block oriented models have been developed for single-input-single-output (SISO) systems. The main limitation of these approaches is that the order of the dynamics and the non-linearities are assumed to be known *a priori*.

In this paper, both Hammerstein and Wiener type models are proposed for describing non-linear systems. The Hammerstein model using CVA, based on the work of Lakshminarayanan et al. [1], is first reviewed. The methodology is extended to select without any *a priori* knowledge the order of both the dynamic and non-linear element. A novel Wiener type model is then proposed based on a state space representation. The linear dynamic element is approximated by the subspace projection technique of CVA, whilst the static non-linear block is calculated using polynomial functions. The approach is multivariable, with all model parameters being calculated without any *a priori* knowledge of the system orders. Both Hammerstein and Wiener type of models and solution algorithms are validated and compared using a popular benchmark system, the pH neutralisation process ([5]).

2. Hammerstein Modelling in State Space using Canonical Variate Analysis

The static non-linear element transforms the actual inputs $\mathbf{u(t)}$ to non-linear measures, $\mathbf{u_{nl}(t)}$, whilst the dynamics are modelled by a linear transfer function $\mathbf{G}^*(z^{-1})$ resulting in the calculation of the system output $\mathbf{y(t)}$. Two approaches to calculating the non-linear element have been reported in the literature, separate and combined parameterisations, [6] and [1]. In separate parameterisation, the number of non-linear elements is equal to the number of inputs, n_u, with each non-linear element u_{nl_i} being a weighted sum of the powers of the input u_i:

$$\begin{aligned} u_{nl_1} &= \gamma_{11}u_1 + \gamma_{12}u_1^2 .... + \gamma_{1n_\gamma}u_1^{n_\gamma} \\ &\vdots \\ u_{nl_{n_u}} &= \gamma_{n_u1}u_{n_u} + \gamma_{n_u2}u_{n_u}^2 .... + \gamma_{n_un_\gamma}u_{n_u}^{n_\gamma} \end{aligned} \tag{1}$$

where n_γ is the order of the polynomial and γ_{ij} are the coefficients of the polynomial.

In combined parameterisation, the non-linear element u_{nl} is a weighted sum of the powers and products of all the inputs. For example, given two inputs (n_u=2) and a polynomial of order two, the non-linear input $u_{nl,i}$ is calculated as:

$$u_{nl_i} = \gamma_{i1}u_1 + \gamma_{i2}u_1^2 + \gamma_{i3}u_2 + \gamma_{i4}u_2^2 + \gamma_{i5}u_1u_2 \tag{2}$$

In Eskinat et al. [6] the number of non-linear inputs, $\mathbf{u_{nl}}$, is set equal to the number of system inputs whilst in [1], it is set equal to the number of outputs. Combined parameterisation provides a more powerful tool for capturing the non-linearities in a system than separate parameterisation, but the computational load is higher since more parameters require to be estimated for the same polynomial order.

Various algorithms have been proposed to identify the parameters of a Hammerstein model for single-input-single-output (SISO) systems. Most have been based on the pioneering Narendra-Gallman algorithm (NGA, [2]). Here the linear dynamic element and the non-linear gain polynomial are updated separately and sequentially. A major drawback with all the existing parametric methods is that the order of the dynamic part is assumed to be known *a priori*. Lakshminarayanan et al. [1] proposed a Hammerstein model identification procedure based on Canonical Variate Analysis (CVA). They made use of the iterative algorithm of Narendra and Gallman for two reasons: (a) robustness to high noise contamination and (b) ease of adaptation for the case where no *a priori* assumption of model order can be made. Lakshminarayanan et al. developed their Hammerstein model using the linear platform of the CVA state space model:

$$\mathbf{x}(t+1) = \mathbf{F}\mathbf{x}(t) + \mathbf{G}\mathbf{u}(t) + \mathbf{w}(t) \tag{3}$$

$$\mathbf{y}(t) = \mathbf{H}\mathbf{x}(t) + \mathbf{A}\mathbf{u}(t) + \mathbf{B}\mathbf{w}(t) + \mathbf{e}(t) \tag{4}$$

where **x**, **y** and **u** are the states, output and inputs of the system, respectively. The noise in the system states and outputs is represented by the terms **w** and e, which have covariance matrices **Q** and **R** respectively. The state space model matrices **F**, **G**, **H** **A** and **B** and the noise covariance matrices can be calculated using a least squares solution if the system states, **x**, are known.

The approximation of the system states using Canonical Variate Analysis (CVA) was proposed by Larimore [7] who introduced the notation of the past, $\mathbf{p}$, and the future, $\mathbf{f}$, of a process at time t:

$$\mathbf{p}(t) = \left[\mathbf{y}^T(t-1), \mathbf{y}^T(t-2), \ldots \mathbf{u}^T(t-1), \mathbf{u}^T(t-2), \ldots\right]^T \tag{5}$$

$$\mathbf{f}(t) = \left[\mathbf{y}^T(t), \mathbf{y}^T(t+1), \ldots\right]^T \tag{6}$$

The past vector, $\mathbf{p}$, may include past values of both inputs and/or outputs associated with time t. The future vector, $\mathbf{f}$, includes future values of the outputs associated with time t. The canonical variables corresponding to the vectors $\mathbf{p}$ and $\mathbf{f}$ are calculated such that the correlation between them is maximised. The computational steps are as follows. SVD is first performed on the product of the covariance matrices:

$$\mathbf{\Sigma}_{pp}^{-1/2}\mathbf{\Sigma}_{pf}\mathbf{\Sigma}_{ff}^{-1/2} = \mathbf{V}_1\mathbf{S}\mathbf{V}_2^T \tag{7}$$

The past canonical variables are then given by:

$$\mathbf{x}_p = \mathbf{V}_1^T\mathbf{\Sigma}_{pp}^{-1/2}\mathbf{p} = \mathbf{J}\mathbf{p} \tag{8}$$

where **J** is the canonical variate transformation matrix. The canonical variables, $\mathbf{x_p}$, can then be used to approximate the true system states:

$$\hat{\mathbf{x}} = \mathbf{x}_p \tag{9}$$

The matrices **F**, **G**, **H** and **A** in equations (3) and (4) represent the system dynamics. The transfer function of this state space model is given by:

$$\mathbf{G}^*(z) = \mathbf{H}(z\mathbf{I} - \mathbf{F})^{-1}\mathbf{G} + \mathbf{A} \tag{10}$$

In the Hammerstein model, it is assumed that the non-linearities enter the system through the static non-linear functions of the inputs. To develop the CVA-Hammerstein model, the actual inputs are substituted by their non-linear functions $\mathbf{u_{nl}}$. For separate parameterisation, the coefficients γ_{ij} of the polynomial of the inputs are calculated as follows. The non-linear element can be written in matrix format:

$$\mathbf{u}_{nl}^T(t) = \begin{bmatrix} \mathbf{u}_1^*(t) & 0 & 0 & 0 & 0 \\ 0 & \mathbf{u}_2^*(t) & 0 & \cdots & 0 \\ 0 & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & 0 & \mathbf{u}_{n_u}^*(t) \end{bmatrix} \mathbf{\Gamma} = \mathbf{U}^*(t)\mathbf{\Gamma} \tag{11}$$

where $\mathbf{u}_i^*(t)$ $(1 \times n_\gamma)$ and $\mathbf{\Gamma}$ $(n_u p \times 1)$ are given by:

$$\mathbf{\Gamma}^T = \left[\gamma_{11} \quad \cdots \quad \gamma_{1p} \quad \ddots \quad \gamma_{n_u 1} \quad \cdots \quad \gamma_{n_u n_\gamma}\right] \tag{12}$$

$$\mathbf{u}_i^*(t) = \left[u_i(t) \quad u_i^2(t) \quad \cdots \quad u_i^{n_\gamma}(t)\right] \tag{13}$$

Combining equations (10) to (13) gives:

$$\mathbf{y}(t) = \mathbf{G}^*(z)\mathbf{U}^*(t)\boldsymbol{\Gamma} \text{ or } \mathbf{y}(t) = \mathbf{C}(t)\boldsymbol{\Gamma} \tag{14}$$

where $\mathbf{C}(t)$ is given by:

$$\mathbf{C}(t) = [\mathbf{H}(z\mathbf{I} - \mathbf{F})^{-1}\mathbf{G} + \mathbf{A}]\mathbf{U}^*(t) \tag{15}$$

Equation (14) represents a linear system of equations with the well known least squares solution:

$$\boldsymbol{\Gamma} = \left[\mathbf{C}(t)^T\mathbf{C}(t)\right]^{-1}\left[\mathbf{C}(t)^T\mathbf{y}(t)\right] \tag{16}$$

Since the non-linear element, $\mathbf{u}_{nl}$, is known, the system matrices **F**, **G**, **H** and **A** can be updated using the last value of $\boldsymbol{\Gamma}$. The CVA-Hammerstein algorithm can be summarised by the following computational steps:

1. Construct a linear CVA model between the actual inputs, **u**, and outputs, **y**. Obtain matrices **F**, **G**, **H** and **A**.
2. Calculate $\boldsymbol{\Gamma}$ from equation (16)
3. Calculate the non-linear element $\mathbf{u}_{nl}$ from (11).
4. Obtain better estimates of the state space model matrices using $\mathbf{u}_{nl}$ instead of the actual inputs.
5. Check for convergence. If convergence occurs, stop, otherwise go to step 2.

For convergence, Lakshminarayanan et al. [1] normalised the coefficients γ_i of each input by dividing by the maximum coefficient, $\max(\gamma_{ij})$. The Euclidean norm of the difference of the γ_{ij} coefficients of the current and the previous step is calculated and is driven to a value of 10^{-7}.

The parameters to be estimated in CVA-Hammerstein modelling are the system dynamics, K, i.e. the order of the past vector, the order of the state vector, k, and the order of the polynomial, n_γ. Lakshminarayanan et al. [1] proposed using *AIC* to select the order of the past vector based on the Augmented Upper Diagonal Identification algorithm, [9]. This procedure compares the values of AIC from the development of AutoRegressive with eXogeneous variables (ARX) models for increasing orders of the past vector. The major drawback of the approach in [1] is that the same time lag is assigned to each input and output included in the past vector. Thus, a restriction is imposed in terms of capturing the true system dynamics. To overcome this problem, Simoglou et al. [10] proposed using *AIC* in a similar way but allowing each input and output to have a different time lag. In this paper, the development of an ARX model is used to identify the order of the past vector. However, a wide variety of information and fit-error criteria are applied including, Akaike Information Criterion (*AIC*), Final Prediction Error (*FPE*), Bayesian Information Criterion (*BIC*), Law of Iterated Logarithms Criterion (*LILC*), Adjusted Multiple Correlation Coefficient (R_a^2) and Overall F-test of the Loss Function (*OVF*).

The state order k, and the polynomial order n_γ, are calculated separately and sequentially in accordance with NGA. For a given state order, k, various Hammerstein models are developed for increasing polynomial order, n_γ. The optimal model is then selected by consulting the above mentioned criteria.

3. Wiener Modelling in State Space using Canonical Variate Analysis

Various non-linear system representations have been proposed based on the framework provided by the Wiener model. Norquay et al. [11] proposed a SISO Wiener model where the linear element was approximated by either an ARX or a step-response model. The non-linear element was approximated by a polynomial function. The SISO Wiener model was then applied to develop a Model Predictive Control (MPC) scheme for a pH neutralisation process. Patwardhan et al. [4] developed Hammerstein and Wiener multi-input multi-output (MIMO) models using the statistical framework of partial least squares (PLS). They described the non-linear behaviour of the system by identifying the PLS inner relationship between the input and the output PLS latent variables using Hammerstein and Wiener Models. The linear dynamic element was approximated by an ARX model whilst the non-linear element was approximated using a polynomial function.

In this paper, a novel Wiener model is proposed. The intermediate transformed variables (output of the linear dynamic element) are calculated using CVA. A state space model is then developed where the states of the system are approximated by polynomial functions of the intermediate transformed variables. The whole identification procedure is automated with the various orders of the system being calculated using well-known criteria.

The state space representation for developing the Wiener model is of the following form:

$$\mathbf{x}(t+1) = \mathbf{F}\mathbf{x}(t) + \mathbf{w}(t) \text{ with } \mathbf{w}(t) = \mathbf{B}\mathbf{e}(t) \tag{17}$$

$$\mathbf{y}(t) = \mathbf{H}\mathbf{x}(t) + \mathbf{e}(t) \tag{18}$$

The state space and noise covariance matrices are computed using least squares. The states of the system are approximated using CVA. CVA calculates as intermediate variables, **x**, the states that are linear combinations of

past values of the inputs and/or the outputs, with the total number of states being k. At this stage, the linear dynamic element of the Wiener model is identified. The non-linear element is then developed by calculating non-linear states that are polynomial functions of the linear states obtained by CVA according to the separate parameterisation technique presented in section 2. In matrix format, this is given by:

$$\mathbf{x}_{nl}^{T}(t)=\begin{bmatrix}\mathbf{x}_1^*(t) & 0 & 0 & 0 & 0\\ 0 & \mathbf{x}_2^*(t) & 0 & \dots & 0\\ 0 & 0 & \ddots & \ddots & \vdots\\ \vdots & \vdots & \ddots & \ddots & 0\\ 0 & 0 & \cdots & 0 & \mathbf{x}_k^*(t)\end{bmatrix}\boldsymbol{\Gamma}=\mathbf{X}^*(t)\boldsymbol{\Gamma} \tag{19}$$

where $\mathbf{x}_i^*$ $(1\times k)$ and Γ $(kp\times 1)$ are given by:

$$\boldsymbol{\Gamma}^T=\begin{bmatrix}\gamma_{11} & \cdots & \gamma_{1p} & \ddots & \gamma_{k1} & \cdots & \gamma_{kn_\gamma}\end{bmatrix} \tag{20}$$

$$\mathbf{x}_i^*(t)=\begin{bmatrix}x_i(t) & x_i^2(t) & \cdots & x_i^{n_\gamma}(t)\end{bmatrix} \tag{21}$$

The above non-linear states replace the original linear states in the state space model:

$$\mathbf{y}(t)=\mathbf{H}\mathbf{x}_{nl}(t)+\mathbf{e}(t) \tag{22}$$

The advantage of the proposed model is that it introduces the non-linearities not in the static inputs but in the dynamic states, which contain the important past system information. To calculate the coefficients γ_{ij} of the polynomial in equation (20) a computational procedure, similar to that of the Hammerstein approach, is followed. Equation (22) can be written as :

$$\mathbf{y}(t)=\mathbf{H}\mathbf{x}_{nl}(t)+\mathbf{e}(t)=\mathbf{H}\mathbf{X}^*(t)\boldsymbol{\Gamma}+\mathbf{e}(t)=\mathbf{C}\boldsymbol{\Gamma}+\mathbf{e}(t)\quad\text{where}\,\mathbf{C}=\mathbf{H}\mathbf{X}^*(t) \tag{23}$$

The matrix of the polynomial coefficients, Γ, is then obtained by the least squares solution:

$$\boldsymbol{\Gamma}=\left[\mathbf{C}(t)^T\mathbf{C}(t)\right]^{-1}\left[\mathbf{C}(t)^T\mathbf{y}(t)\right] \tag{24}$$

The Wiener state space algorithm involves four computational steps:

1. Calculate the transformation matrix, $\mathbf{J}$ that transforms the past vector, $\mathbf{p}$, to the linear system states using CVA.
2. Calculate the state space model matrices, $\mathbf{F}$ and $\mathbf{H}$.
3. Obtain the coefficients γ_{ij} using equation (24).
4. Check if Γ converges. If yes, stop, otherwise go to step 2. The same convergence criterion as described for Hammerstein modelling is again used.

As for Hammerstein modelling, the parameters to be estimated in Wiener state space modelling are the system dynamics, K (i.e. the order of the past vector), the order of the state vector, k, and the polynomial order, n_γ.

Patwardhan et al. [4] in their PLS approach do not mention how they select the various system orders and they do not give details of the Wiener model algorithm. They refer to [3] where only the PLS Hammerstein model was originally proposed. However, in [3] no specific criteria were mentioned for model order selection. Norquay et al [11] in their SISO Wiener approach calculated the order of the linear dynamic element (ARX or step-response model) using the forward-regression orthogonal estimator [12] and the method of Lipschitz numbers [13], whilst no specific procedure was suggested to select the order of the non-linear element. In this paper, the order selection procedures applied in Hammerstein modelling are applied to Wiener modelling as well.

4. Case Study of a pH Neutralisation Process

A simulation of a pH neutralisation process is used to validate and compare the previously described methods ([5]). The process was selected since it is characterised by stiff dynamics and strong non-linearities. The process consists of a tank where an acid (HNO_3) is neutralised by a strong base (NaOH). A reference (6000 samples) and a validation (1000 samples) data set were generated to build and validate the various models. To make the comparison more robust various levels of noise were used to investigate the behaviour of the various models in the presence of noise. Random white noise was added to the previously generated reference and validation data sets. Three signal to noise ratios were considered, S/N=10, 3 and 1. The range of pH (3.5 to 10.5) was selected to be wide enough so that the data was subject to strong non-linearities and to make the identification task more challenging. The wider the range of pH, the greater the non-linearity. To predict the system output (pH) two inputs were considered (acid and base flow rates).

The results are sumarised in Table 1. For each model, the first column (M_k) shows the number of identified parameters, i.e. total number of state space parameters and polynomials, respectively. The second and third column show the Mean Squared Error (MSE) for the reference and the validation data set. The residuals used in the calculation of the MSE are defined to be the difference between the model predictions and the actual values without noise. Therefore, the metrics, $MSE_{r,1}$ and $MSE_{v,1}$, show not only how well the various methods can predict the data but also how well they filter the noise. The last two columns are calculated in the same way as the previous two. The only difference is that the residuals are calculated as the difference between the model predictions and the actual noisy data values. From inspection of the results, the following conclusions can be drawn:-

1 Wiener modelling provided better results than either types of Hammerstein modelling in terms of lower Mean Squared Error, for the data where the S/N was 10 and 3. However, for a S/N equal to 1 the results for the two model types become comparable. The superiority of the Wiener modelling stems from the way it describes the non-linearities in the data. In contrast to Hammerstein modelling, the Wiener models introduce the non-linearities in the dynamic part, i.e. the states that include all the important information from the past inputs and outputs of the process. In contrast, the Hammerstein model describes the non-linear behaviour of the data by calculating static non-linear functions of the current process inputs.

2 In general, combined parameterisation deals with the system non-linearities in a more advanced way than separate parameterisation. However, if the static non-linearity is of simple structure, then both types of parameterisation are expected to exhibit similar performance. This is the reason why combined and separate parameterisation performs in a similar manner for all levels of noise. However, combined parameterisation always provides more accurate predictions.

3 Model predictions become worse as the level of noise increases in the data. $MSE_{r,1}$ and $MSE_{v,1}$ values are always much smaller than $MSE_{r,2}$ and $MSE_{v,2}$ v respectively. This means that the distance of the predictions from the noise free data is smaller than that from the filtered data. Thus, all models manage, to some degree, to filter the noise. This was expected, since by projecting the data onto new lower dimensional spaces, CVA excludes the noise from the data. Examples of how the noise is filtered through CVA Wiener modelling is given in Figure 1 and Figure 2 where the validation predictions are plotted along with the actual and noisy values for a S/N=3 and 1 respectively.

Table 1: Results for Hammerstein Modelling using Separate Parameterisation

	Hammerstein Model Separate Parameterisation					Hammerstein Model Combined Parameterisation					Wiener Model				
S/N	M_k	$MSE_{r,1}$	$MSE_{v,1}$	$MSE_{r,2}$	$MSE_{v,2}$	M_k	$MSE_{r,1}$	$MSE_{v,1}$	$MSE_{r,2}$	$MSE_{v,2}$	M_k	$MSE_{r,1}$	$MSE_{v,1}$	$MSE_{r,2}$	$MSE_{v,2}$
10	15	0.108	0.103	0.141	0.143	20	0.107	0.102	0.141	0.142	179	0.073	0.071	0.106	0.111
3	15	0.204	0.199	0.557	0.528	20	0.203	0.197	0.557	0.526	67	0.160	0.151	0.518	0.486
1	15	0.603	0.630	3.917	3.792	16	0.603	0.628	3.917	3.790	19	0.614	0.626	3.917	3.766

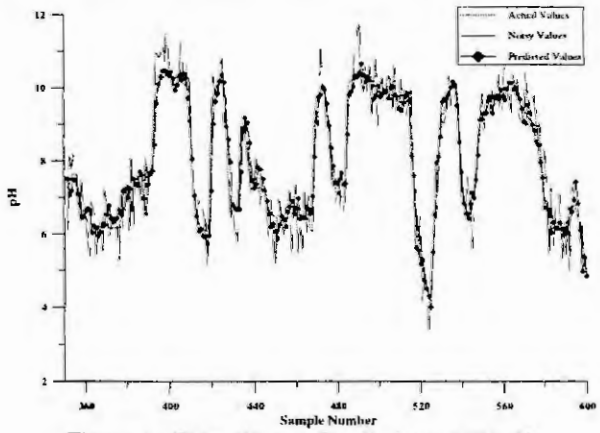


Figure 1: CVA Wiener Predictions (S/N=3)

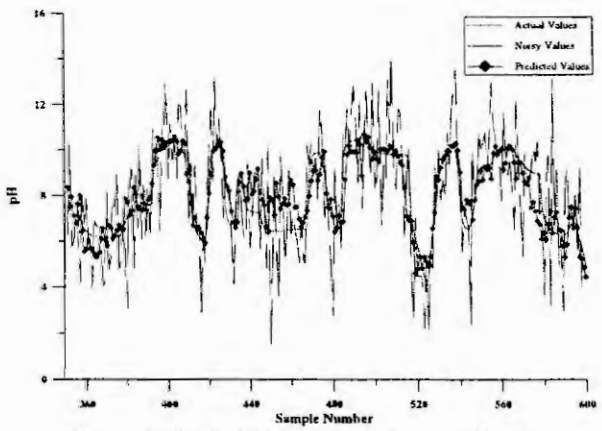


Figure 2: CVA Wiener Predictions (S/N=1)

5. Conclusions

In this paper the issue of non-linear system identification was investigated. The Hammerstein modelling was based on the work of Lakshminarayanan et al. [1] In this paper, their approach has been extended to allow the selection of the order of the non-linear block in contrast to previous approaches where selection was based on *a priori* knowledge. Moreover, additional information and error criteria such as the Bayesian Information Criterion (BIC) and Adjusted Multiple Correlation Coefficient (R_a^2), were considered to realise more reliable order selection.

A new approach for developing Wiener type models was also proposed. The dynamic linear block was approximated using the subspace projection technique of CVA. The static non-linear element was then calculated as a polynomial function of the linear states using the separate parameterisation. The approach is multivariate with all the parameters are calculated without any *a priori* knowledge of the orders of the system.

Both Hammerstein and Wiener type models and solution algorithms were compared using a benchmark pH neutralisation process. It was found that the proposed Wiener model was superior to the Hammerstein approach in terms of providing more accurate predictions for signal to noise ratio up to three. The only disadvantage of the Wiener model is that it requires the identification of more parameters.

6. Acknowledgements

The authors acknowledge the support of the EPSRC IMI Project, System Capability Enhancement in High Perfomance Monitoring (SCIENTIA) and the EPSRC/DTI Project, Multivariate Statistical Performance Monitoring and Sensor Management (MENTOR).

7. References

1. Lakshminarayanan, S., S.H. Shah, and K. Nandakumar, Identification of Hammerstein Models using Multivariate Statistical Tools. Chemical Engineering Science, 1995. 50(22), 3599-3613.
2. Narendra, K.S. and P.G. Gallman, An Iterative Method for the Identification of Nonlinear Systems using the Hammerstein Model. IEEE Transaction on Automatic Control, 1966. 2, 546-550.
3. Lakshminarayanan, S., S.L. Shah and K. Landakuram, Modeling and Control of Multivariable Processes: Dynamic PLS Approach. AIChE Journal, 1997. 43(9), 2307-2322.
4. Patwardhan, R.S., S. Laksminarayanan, and S.L. Shah, Constrained Nonlinear MPC Using Hammerstein and Wiener Models: PLS Framework. AIChE Journal, 1998. 44(7), 1611-1622.
5. Henson, M.A. and D.E. Seborg, Adaptive Non-linear Control of a pH Neutralisation Process. IEEE Transactions on Control Systems Technology, 1994. 3, 169-183.
6. Eskinat, E., S.H. Johnson, and W.L. Luyben, Use of Hammerstein Models in Identification of Nonlinear Systems. AIChE Journal, 1991. 37(2), 255-268.
7. Larimore, W.E., System identification, Reduced Order Filtering and Modeling via Canonical Correlation Analysis. Proceedings of the American Control Conference, 1983, 445-51.
8. Haist, N.D., F.H.I. Chang, and R. Luus, Nonlinear Identification in the Presence of Correlated Noise using a Hammerstein Model. IEEE Transaction on Automatic Control, 1973. 18, 552-555.
9. Niu, S. and D.G. Fisher, Simultaneous Structure Identification and Parameter Estimation of Multivariable Systems. International Journal of Control, 1994. 59(5), 1127-1141.
10. Simoglou, A., Martin, E. B. and Morris, A. J. Dynamic Multivariate Statistical Process Control using Partial Least Squares and Canonical Variate Analysis, Computers Chem Engng, 1999. 23, S277-S280.
11. Norquay, S.J., A. Palazoglou, and J.A. Romagnoli, Model Predictive Control on Wiener Models. Chemical Engineering Science, 1998. 53(1), 75-84.
12. Billings, S.A., S. Chen, and M.J. Korenberg, Identification of MIMO Non Linear Systems using a Forward-Regression Orthogonal Estimator. International Journal of Control, 1989. 49, 2157-2189.
13. He, X. and H. Asada. A New Method for Identifying Orders of Input-Output Models for Non-Linear Dynamic Systems. American Control Conference. 1993. San Fransisco.

ALGORITHM DETERMINING THE SLIDING WINDOW CONTAINING THE CHANGE POINT IN ARMA PARAMETERS

R. Souidi[1] and A. Guesbaoul[1,2]

[1] Université des Sciences et de la Technologie d'Oran . U.S.T.O.

BP 1505, El m'naouer 31000 Oran, Algeria.

[2] I.N.P.L. E.N.S.E.M , Dea Atns ; 2, Avenue de la foret-de-Haye

54516 Vandoeuvre-les-Nancy , France.

Abstact. The extention of the sequentiel detection of the rupture point for the case of the MA and ARMA process proposed by Basseville and Benveniste [1] seeming impossible to be used, we suggest [5], a detection strategy from a mobile fixed sized window. However, the limits of the statistic used to test the position of the sliding window itself were not exactly etablished. Despite this, as shown by simulations [7], the test statistic still provided good approximations using the χ^2 law. In the present paper, we develop this statistic through another, whose limit law is a χ^2, and justify the former statistic's approximations.

Introduction.

Let $(X_t\ ;\ t \in Z)$ be a real and Gaussian stochastic process satisfying an ARMA(p,q) model :

$$\sum_{i=0}^{p} a_i\, X_{t-i} = \sum_{j=0}^{q} b_j\, \epsilon_{t-j}\ ; \quad a_0 = b_0 = 1$$

Let : $\theta = (\, a_1, ..., a_p, b_1, ..., b_q\,)^t$ being the parameter vector and the perturbations ϵ_t assumed to be normally distributed with mean zero and variance σ^2. We may write :

$$X_t = \sum_{j=0}^{\infty} m_j^{\theta}\, \epsilon_{t-j}\ ; \quad m_0^{\theta} = 1\ . \qquad \epsilon_t = \sum_{j=0}^{\infty} d_j^{\theta}\, X_{t-j}\ ; \quad d_0^{\theta} = 1\ .$$

We note : $h_\theta(z) = \sum_{j=0}^{\infty} m_j^{\theta}\, z^j$ and $h_\theta^{-1}(z) = \sum_{j=o}^{\infty} d_j^{\theta}\, z^j$.

Let : $\eta(t,\theta) = \sum_{t-j \in obs(t)} d_j^{\theta}\, X_{t-j}$, where : $obs(t) = \{1, ..., n\} \cap \{s \in Z\ ,\ s \le t\}$; if θ_n^i is an initial estimator of θ, we recall that the function :

$$L_n^{B,J}(x, \theta_n^i) = -\tfrac{n}{2} \operatorname{Log} 2n\sigma^2 - \tfrac{1}{2\sigma^2} \sum_{t=1}^{n} \eta^2(t, \theta_n^i)\ .$$

is the approximation of the likelihood function given by Box Jenkins [2] and the function given by Whittle [3] :

$$L_n^{W}(x, \theta_n^i) = -\tfrac{n}{2} \operatorname{Log} 2n\sigma^2 - \tfrac{1}{4\pi} \int_{-\pi}^{\pi} f^{-1}(\theta_n^i, \lambda)\, I_n(\lambda)\, d\lambda\ .$$

where the spectral density $f(\theta, \lambda)$ of X_t is written : $f_\theta(\lambda) = \frac{\sigma^2}{2\pi} \left| h_\theta(e^{i\theta}) \right|^2$ and $I_n(\lambda)$ the periodogramm of X_t : $I_n(\lambda) = \frac{1}{2\pi n} \sum_{t=1}^{n} X_t\, e^{i\lambda t} \sum_{t=1}^{n} X_t\, e^{-i\lambda t}$.

We note : $G_n(\theta) = \frac{1}{n} \sum_{t=1}^{n} \eta^2(t, \theta)$

and $s_{n,j}(\theta) = \frac{\partial}{\partial \theta_j} \left(-n^{-1} L_n^{B,J}(\theta) \right)$; $j = 1, ..., p+q$.

We recall that :

$$p\text{-}\lim_{n\to\infty} s_{n,j,k}(\theta_n) = p\text{-}\lim_{n\to\infty} \left[\frac{\partial^2}{\partial\theta_j \partial\theta_k} \left(-n^{-1} L_n^{B,J}(\theta_n) \right) \right] = S_{j,k}\ ; \quad j,k = 1, ..., p+q$$

where : $S_{j,k} = \frac{1}{\sigma^2} \int_{-\pi}^{\pi} \frac{\partial}{\partial\theta_j} h_\theta^{-1}(e^{i\lambda})\, f(\lambda, \theta)\, \frac{\partial}{\partial\theta_k} h_\theta^{-1}(e^{i\lambda})\, d\lambda$.

and θ_n is a consistent estimator of θ_0 i.e : $p\text{-}\lim_{n\to\infty} \theta_n = \theta_0$ [4] .

We now consider the question of testing the null hypothesis of change, $H_0 : \theta = \theta_0$, against H_1 : there exists $n^* \in Z$ such that : $r \in (n^* - k, n^* - k+1, ..., n^*)$ where r is the change point defined by : $\theta = \theta_0$ if $t \leq r$; $\theta = \theta_1$ if $t \succ r$.

$x_1, x_2, ..., x_n, ...x_N$. being observed , we bave used the maximum of likelihood method, but this method needs the computation of N reports corresponding to the possible values of r, which is expensive. So, we propose an algorithm elaborated using a mixed method. At first, we fixe the window and then we use the likelihood report to estimate r among the k possibilities of the chosen window (k being the length of the window).With the help of the techniques based upon the maximum of likelihood method, we have built two estimators of θ_0, let θ_{n-k} and θ_n from observations $(x_1,....,x_{n-k})$ and $(x_1,...,x_n)$ repectively, according to the formula :

$$\theta_n = \theta_{n-k} - S_n^{-1} s_n (\theta_{n-k}) .$$

We start by moving the sliding window in such a way that observations $(x_1,....,x_{n-k})$ allow a θ_{n-k} estimation fairly close to θ_0 to be calculated. Thus :

$$\forall \varepsilon \succ 0, \forall n \geq n_0 \in N , P (\theta_{n-k} \in B (\theta_0 , \varepsilon)) = 1 - \varepsilon .$$

which indicates that, the point change , if any there may be , only occurs once the θ_{n-k} estimation of θ_0 is stable, i.e. that : $n_0 \prec r \prec N$.

To determine the position of the sliding window, we compare the θ_n and θ_{n-k} estimators. If they do not differ from each other significantly, we reject H_1, and increase n in steps of 1. If, on the other hand, θ_n and θ_{n-k} do differ from each other significantly, we reject H_0, and decide that the change point is located within the sliding window $(n^* - k , n^*)$.

Main results

Under H_0, assuming that X_t satisfies the hypothesis of stationarity, ergodicity and regularity, the following applies :

if $M_n = \sqrt{n} [G_n (\theta_n) - G_n (\theta_{n-k})]$; we show that : $p - \lim_{n \to \infty} M_n = 0$.

We introduce the function : $F_n (\theta) = n (\theta - \theta_{n-k})^t S_n (\theta - \theta_{n-k})$ and we give the following theorem :

Theorem : Under the hypothesis H_0, the statistic : $R_n = \sup (F_n (\theta_n) ; \theta_n \in V(\theta_{n-k}))$, where $V (\theta_{n-k})$ is the neighbourhood of θ_{n-k} ; exists and verifies :

$R_n = F_n (\theta_0)$. Hence, R_n is asymptotically distributed as χ^2 with p+q df .

The test

H_0 is rejected as soon as : $R_n \succ t_\alpha$ (1), where t_α is the critical thresold of the χ^2 law with p+q df corresponding to the given level of significance α . However, constructing R_n is not simple. A sufficient condition for satisfying (1) is :

$$F_n (\theta_n) = Z_n^t S_n Z_n \succ t_\alpha \qquad (2)$$

In practice [6], to reduce the delay in detection, it is possible to choose the critical region of :

$$Z_n^t S_n Z_n + M_n \succ t_\alpha \qquad (3)$$

Moreover, if n^* is the first n which verifies inequality (3) :

we decide that : $r \in (n^* - k , n^*)$.

Parameters estimations under hypothesis H_1.

Let θ_r^0 (resp.θ_r^1) the estimaor of θ_0 (resp.θ_1) obtained from the set of observations $(x_1, ..., x_{n*-k})$ (resp. $(x_{n*},...,x_n)$) .

Now, let $l \in (n^* - k + 1, ..., n^* - 1)$; we note θ_l^0(resp.θ_l^1) the estimator of θ_0(resp.θ_1) obtained from the set of observations $A = (x_1, ...,x_l)$ (resp. $B = (x_{l+1}, ...,x_n)$).

We write then :

on the set A : $\theta_l^0 = \theta_r^0 - S_l^{-1} (\theta_r^0) s_l (\theta_r^0)$

and on the set B : $\theta_l^1 = \theta_r^1 - S_{n-l}^{-1} (\theta_r^1) s_{n-l} (\theta_r^1)$.

The estimator r^* of r is obtained bt minimizing the expression :

$$\lambda_l = {}^t (\theta_{l-1}^0 - \theta_r^0) (\theta_{l-1}^0 - \theta_r^0) + {}^t (\theta_{l+1}^1 - \theta_r^1) (\theta_{l+1}^1 - \theta_r^1)$$

and we write : $r^* = \arg\min (\lambda_l)$.

Simulation

We consider the simulation [6], of a first order MA using 1000 observations under the following model :

$$X_t = \begin{cases} \theta_0\, \varepsilon_{t-1} + \varepsilon_t & \text{if} \quad 1 \le t \le r \\ \theta_1\, \varepsilon_{t-1} + \varepsilon_t & \text{if} \quad r+1 \le t \le 1000 \end{cases}$$

where r = 600 , θ_0 = -0.4 ; θ_1 = 0.9 ; σ^2 = 1 and k = 200.

using the critical region gefined by (3), and with α = 0.05; the χ^2 table with 1 df gives t_α = 3.841. The simulation results are made out on computer DPS8/70. We obtain, for five calculations, the following values of n* :

n^*	θ_r^0	θ_r^1	r^*	$\theta_{r^*}^0$	$\theta_{r^*}^1$
699	-0.408	0.844	551	-0.433	0.849
712	-0.341	0.902	582	-0.358	0.902
796	-0.387	0.903	599	-0.388	0.901
749	-0.461	0.915	606	-0.460	0.913
641	-0.341	0.896	600	-0.402	0.894.

References

1. Basseville,M. and Benveniste, A., Detection sequentielle de changements brusques des caractéristiques spectrales d'un signal numérique.Rapport de recherche INRIA, Rennes, 129 (1982).

2. Box, G.E.P. and Jenkins, G.M., Time series analysis. Holden day, 1976.

3. Whittle, Gaussian estimation in stationnary time series. Bull. Int. Stat. Inst., 33 (1967).

4. Pham Dinh, Estimation et test dans les modèles de processus stationnaires. Thèse d'etat, Grenoble, 1975.

5. Souidi, R., On the distribution of the test statistic for detecting a point of change in real and Gaussian ARMA parameters. Applied Mathematics and letters,2(1989),207-210.

6. Souidi, R., A new sequentiel test for detection of a point of change in real ARMA parameters. Computers Mathematics Applications, 19 (1990), 31-39.

7. Guesbaoui, A., Détection par simulation de points de ruptures dans les modèles ARMA.non stationnaires. Inst. Infor. U.S.T.O., Oran, 1996.

ORTHOGONAL ECLMS ALGORITHM FOR DOUBLE-TALK ECHO CANCELLING

K. Yamashita, A. Shimabukuro, M. R. Asharif, H. Miyagi

Fculty of Engineering, University of the Ryukyus
1-Senbaru, Nishihara, Okinawa, 903-0213, Japan
yamasita@eee.u-ryukyu.ac.jp

Abstract In order to solve the double-talk problem in the echo cancelling for teleconference system, the authors have proposed the extended correlation least mean squares (ECLMS) algorithm. However, this algorithm does not give a fast convergence. The purpose of this paper is to derive an orthogonal ECLMS with the lattice strucure in order to speed up the convergence characteristics and to reduce the computational load for the adaptation.

Introduction

In hands-free set mobile radiotelephone or teleconference systems, where acoustic feedback exists between the loudspeaker and microphone, the quality of communication is degraded severely, because of the acoustic echo impulse response of the car or the teleconference room. Then the adaptive finite impulse response (FIR) filters with the least mean squares (LMS) and the normalized least mean squares (NLMS) algorithms [3] are utilized for the echo cancelling. However, in the double-talk environment when both the near-end and the far-end signals exist at the same time, these algorithms do not give a sufficient performance.

In order to obtain an acceptable performance even in the presence of the double-talk, the authors proposed the correlation least mean squares (CLMS) algorithm [1], in which the gradient was obtained from the correlation function of the input signal instead of the input signal. This algorithm gives a better performance than the NLMS algorithm, however, it does not present a sufficient performance yet. In order to overcome the weak point, we have proposed the Extended CLMS (ECLMS) algorithm [2]. In this algorithm, a sum of lagged squared errors has been considered as the cost function. Then, this algorithm shows a better performance compared with the CLMS algorithm but still does not give a fast convergence characteristics.

The purpose of this paper is to derive an orthogonal ECLMS algorithm with the lattice strucure in order to speed up the convergence characteristics and to reduce the computational load for the adaptation. The validity of this algorithm is evaluated by computer simulations in the echo cancelling problem.

Theory

ECLMS Algorithm

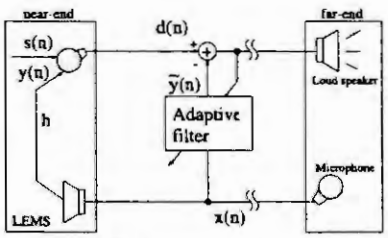


Fig.1. Echo cancelling system.

First of all, we present the ECLMS algorithm [2] for the teleconference system shown in Fig.1. The correlation functions for the input signal $x(j)$ called the far-end signal are defiened by

$$\phi_{xx}(n,k) = \sum_{j=0}^{n} x(j)x(j-k) \tag{1}$$

and the correlation functions between the desired signal $d(j)$ and $x(j)$ are defined by

$$\phi_{dx}(n,k) = \sum_{j=0}^{n} d(j)x(j-k) \tag{2}$$

where

$$d(j) = s(j) + y(j)$$

with

$$y(j) = \sum_{i=0}^{N-1} h_i x(j-i)$$

Here $s(j)$ and $y(j)$ show the near-end and the echo signals, respectively, and h_i is the impulse response of the loudspeaker enclosure microphone system (LEMS). Since $s(j)$ and $x(j-k)$ are two independent speech signals, the correlation between $s(j)$ and $x(j-k)$ is almost zero. Therefore, the correlation function $\phi_{dx}(n,k)$ between $d(j)$ and $x(j-k)$ signals becomes

$$\phi_{dx}(n,k) \simeq \sum_{i=0}^{N-1} h_i \phi_{xx}(n,k-i) \tag{3}$$

Then, on the basis of eqn.(3), the estimated correlation function $\widetilde{\phi}_{dx}(n,k)$ is defined by

$$\widetilde{\phi}_{dx}(n,k) = \sum_{i=0}^{N-1} h_i(n) \phi_{xx}(n,k-i) \tag{4}$$

Next, the cost function is defined by a sum of lagged squared errors as follows:

$$\mathrm{MSE} = \mathrm{E}[\mathbf{e}^{\mathrm{T}}(\mathrm{n})\mathbf{R}\mathbf{e}(\mathrm{n})] \tag{5}$$

where $\mathbf{R}$ is a positive semidefinite diagonal matrix with r_i as weighting factor on the error $e(n,i)$ and

$$\mathbf{e}(n) = [e(n,0), e(n,1), \cdots, e(n,N-1)]^{\mathrm{T}}$$

with

$$e(n,k) = \phi_{dx}(n,k) - \widetilde{\phi}_{dx}(n,k)$$

Using the normalized LMS algorithm to minimize MSE with respect to each coefficient $h_i(n)$, we obtain

$$\mathbf{h}(\mathrm{n}+1) = \mathbf{h}(\mathrm{n}) + \frac{2\mu}{1 + \mathrm{tr}[\mathbf{\Phi}_{\mathrm{xx}}^{\mathrm{T}}(\mathrm{n})\mathbf{R}\mathbf{\Phi}_{\mathrm{xx}}(\mathrm{n})]} \mathbf{\Phi}_{\mathrm{xx}}^{\mathrm{T}}(\mathrm{n})\mathbf{R}\mathbf{e}(\mathrm{n}) \tag{6}$$

where $0 < \mu < 1$, $tr[\cdot]$ means the trace operator and

$$\mathbf{h}(n) = [h_0(n) \quad h_2(n) \quad \cdots \quad h_{N-1}]^{\mathrm{T}}$$

$$\mathbf{\Phi}_{xx}(n) = \begin{bmatrix} \phi_{xx}(n,0) & \phi_{xx}(n,1) & \cdots & \phi_{xx}(n,N-1) \\ \phi_{xx}(n,-1) & \phi_{xx}(n,0) & \cdots & \phi_{xx}(n,N-2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{xx}(n,1-N) & \phi_{xx}(n,2-N) & \cdots & \phi_{xx}(n,0) \end{bmatrix}$$

Orthogonal ECLMS Algorithm

In order to ensure the superior convergence and reduce the computation load for adaptation, we propose an orthogonal ECLMS (OECLMS) algorithm with the lattice structure. First, based on the lattice structure, the forward prediction error $f_i(n)$ and the back prediction error $r_i(n)$ at the stage m are given by

$$\begin{bmatrix} f_i(n) \\ r_i(n) \end{bmatrix} = \begin{bmatrix} 1 & K^i \\ K^i & 1 \end{bmatrix} \begin{bmatrix} f_{i-1}(n) \\ r_{i-1}(n-1) \end{bmatrix} \tag{7}$$

where

$$K^i = -\frac{\mathrm{E}[f_{i-1}(n)r_{i-1}(n-1)]}{\mathrm{E}[f_{i-1}^2(n)]}$$

with the initial condition as

$$f_0(n) = r_0(n) = x(n)$$

where K^i is lattice parameter which is called the reflection coefficient. Then, when the reflection coefficient is set to be optimal in the sense of minimizing the mean square prediction error, the backward prediction errors $r_i(n)(i = 0, \cdots, N-1)$ are orthogonal to each other. This means that the successive stages of the optimal lattice predictor are decoupled from each other at the modeling process. This ensures the optimization of a multistage lattice filter may be accomplished as a sequence of local optimization process at each stage. Furthermore, the transformation of $x(n-i)(i = 0, \cdots, N-1)$ to $r_i(n)(i = 0, \cdots, N-1)$ gives one-to-one correspondence between the far-end signal vector and the backward prediction error vector. Therefore, in spite of $y(n)$ in eqn.(2), we use $y(n)$ with the filter coefficient $\widetilde{h}_i$ given by

$$y(n) = \sum_{i=0}^{N-1} \widetilde{h}_i r_i(n) \tag{8}$$

Similar to the ECLMS case, the correlation functions for the backward prediction error $r_i(n)$ and between the desired signal $d(n)$ and $r_k(n)$ are respectively defiened by

$$\phi_{rr}(n,k) = \sum_{i=0}^{n} r_k^2(i) \tag{9}$$

$$\phi_{dr}(n,k) = \sum_{i=0}^{n} d(i)r_k(i) \tag{10}$$

Since the near-end and the far-end signals are two independent speech signals, $s(n)$ and $r_k(n)$ are independent to each other. Therefore, the correlation between $s(n)$ and $r_k(n)$ is almost zero and the correlation function $\phi_{dr}(n,k)$ between $d(n)$ and $r_k(n)$ signals becomes

$$\phi_{dr}(n,k) \simeq \widetilde{h}_k \phi_{rr}(n,k) \tag{11}$$

Then, on the basis of eqn.eqn(11), the estimated correlation function is defined by

$$\widetilde{\phi}_{dr}(n,k) = \widetilde{h}_k(n)\phi_{rr}(n,k) \tag{12}$$

Next, the cost function is defined by a sum of lagged squared errors as follows:

$$\mathrm{MSE} = \mathrm{E}[\mathbf{e}^{\mathrm{T}}(\mathrm{n})\mathbf{e}(\mathrm{n})] \tag{13}$$

where

$$e(n,k) = \phi_{dr}(n,k) - \widetilde{\phi}_{dr}(n,k)$$

Using the normalized LMS algorithm to minimize MSE with respect to each coefficient $\widetilde{h}_i(n)$, we obtain

$$\widetilde{h}_i(n+1) = \widetilde{h}_i(n) + \frac{2\mu}{1 + \sum_{i=0}^{N-1} \phi_{rr}^2(n,i)} e(n,i)\phi_{rr}(n,i) \tag{14}$$

The rapid convergence for eqn.(14) may be attained, because the orthogonal backward prediction errors are used for adaptation.

Simulation results

To demonstrate the sufficient improvement of the proposed method, we perform the computer simulations which is the comparison of the OECLMS algorithm with the NLMS [3] and the ECLMS [2] algorithms. The echo impulse response h_i is assumed as follows:

$$h_i = Randn[exp(-8i/N)] \tag{15}$$

To measure the performance of the algorithm, we use $D(n)$ defined by the following equation:

$$D(n) = 10\log_{10}\left[E\left[\{y(n)-\tilde{y}(n)\}^2\right]/E\left[\{y(n)\}^2\right]\right] \tag{16}$$

Figure 2 shows the convergence of $D(n)$ for the NLMS, the ECLMS and the OECLMS algorithms in the double-talk situation. The NLMS algorithm hardly converges and is totally blown up in the double-talk situation. The ECLMS algorithm gives a better convergence than the NLMS algorithm but does not give a fast convergence. However, the OECLMS algorithm shows the fast convergence compared with the ECLMS algorithm. In another simulation shown in Fig.3, first starting is the single talk situation, then we change the echo path impulse response after 800 iterations and impose the double-talk situation at the same time and we return the system to the single-talk situation at 1500 iterations. It can be seen from Fig.3 that $D(n)$ of the NLMS algorithm goes up to above 0dB during the double-talk situation, but the ECLMS and the OECLMS algorithms have a satisfactory convergence characteristics in spite of changing the single-talk situation to the double-talk situation. Moreover, the OECLMS algorithm has a fast convergence compared with the ECLMS algorithm.

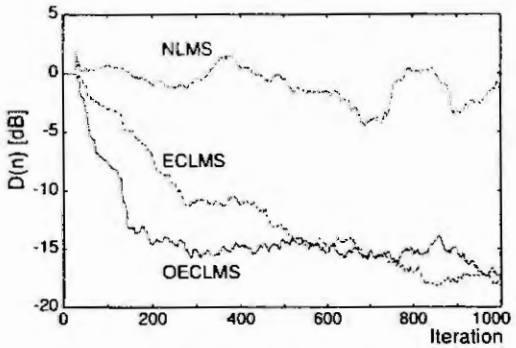


Fig.2. $D(n)$ in the double-talk situation.

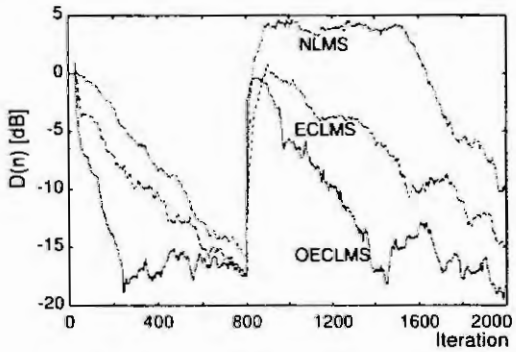


Fig.3. $D(n)$ under changing the echo path and imposing double-talk.

Conclusion

We have presented the OECLMS algorithm which is an expansion to the ECLMS algorithm. In the simulations, we have compared the OECLMS algorithm with the ECLMS and the NLMS algorithms. Then, the sufficient improvement of the proposed algorithm was achieved through computer simulations and we found that this algorithm has fast convergence and robust performance in echo canceller with the double-talk situation, comparing with the NLMS and the ECLMS algorithms. This work was supported by a Grant in Aid for Science Research from the Ministry of Education in Japan.

References

[1] Asharif,M.R., Hayashi,T. and Yamashita,K.,Correlation LMS algorithm and its application to double-talk echo cancelling,Electron. Lett., **35**, (3), 1999, 194-195.

[2] Asharif,M.R., Shimabukuro,A., Hayashi,T. and Yamashita,K.,Expanded CLMS Algorithm for Double-Talk Echo Cancelling, Proc. IEEE SMC'99, Japan, Vol.1, 1999, 998-1002.

[3] Haykin, S.,Adaptive filter theory, Prentice Hall 1991.

A NONLINEAR MODEL FOR RADIAL MAGNETIC BEARINGS

N. Steinschaden and H. Ecker
Vienna University of Technology
Wiedner Hauptstraße 8–10/E303, A–1040 Vienna
http://www.mdmt.tuwien.ac.at

Abstract.

This paper discusses the dynamic characteristics of a single-mass rotor with speed synchronous unbalance excitation supported by an active radial magnetic bearing. The mathematical model includes nonlinear force-to-displacement and force-to-coil current relationships. Nonlinear saturation effects of the magnetic materials as well as limitations of the power amplifier and of the control current are considered in the presented model. Effects of coil inductance combined with an underlying current controller are also taken into account.

Two different numerical approaches are used to solve the system equations. Steady-state solutions are obtained by formulating and solving a boundary value problem, non-periodic solutions by using numerical simulation. For large rotor eccentricities the model shows nonlinear phenomena like coexisting stable and unstable solutions as well as bifurcations and symmetry breaking.

Introduction

Active Magnetic Bearings (AMB) show various significant advantages over conventional bearing types. These advantages have contributed to an increasing interest in this relatively new technology. No need for a lubrication system, no material wear and almost no friction losses are consequences of the contact-free working principle of an AMB, that results in high life span and low maintenance costs. Moreover, stiffness and damping characteristics of an AMB can be modified and adjusted since they are primarily determined by the parameter setting of the feedback control device, see [1]. Consequently AMB are already used in a number of different applications in rotating machinery.

A successful design of a mechatronic system like an AMB is heavily based on mathematical models to study the system behavior in the planning stage and to optimize system parameters. Mainly linear models but also models including certain nonlinearities have been studied and used in the past [4]. However, maximizing the performance of AMB will also result in operational ranges of the bearings that reach further into the nonlinear regime. This creates the need for sophisticated nonlinear mathematical models, including all important nonlinearities as detailed as possible. The presented model should contribute to an increasing understanding of the dynamics of rotating machinery using active magnetic bearings.

Model of the AMB system

Figure 1 shows a schematic diagram of the axial view of an active magnetic bearing. The shaft is plotted in the center position ($x = 0$, $y = 0$, $g = g_0$). The magnetic bearing consists of two pairs of orthogonally oriented actuators (electromagnets). Each pair is controlled independently, based on sensors measuring the rotor position and velocity in the $x-$ and $y-$direction, respectively. These signals are the input signals for the main PID-current controllers. An integral feedback gain eliminates steady state deviations from the center position of the rotor. According to the output of the controller the amplifier supplies the voltage to produce the appropriate magnetic force within the actuator. Figure 2 shows a principle schematic of the feedback control loop of one axis.

Since the model includes the effects of coil inductance it requires additional state variables for the magnetic flux in each actuator. Due to the coil inductance the coil current does not instantanously follow the voltage applied by the power amplifier. Consequently, the voltage drop in each actuator coil has to be calculated. In order to achieve a better dynamic behavior of the magnetic bearing an underlying current controller is implemented. For that task a simple P-controller is employed, see Fig. 2.

Any power amplifier has upper and a lower limits on the output that can be applied to the actuator. If the underlying current controller determines a voltage exceeding the limited voltage range of the amplifier, the required voltage cannot be fully supplied. This results in a reduced slew rate of the coil

currents and consequently in reduced amounts of the magnetic forces, causing poor dynamical behavior of the magnetic bearing.

The rotor (shaft) is represented as a single mass with two degrees of freedom supported by two opposed magnetic actuators in both directions x and y, see Fig. 1. The rotor is excited by speed synchronous unbalance forces, which are caused by the mass imbalance eccentricity e.

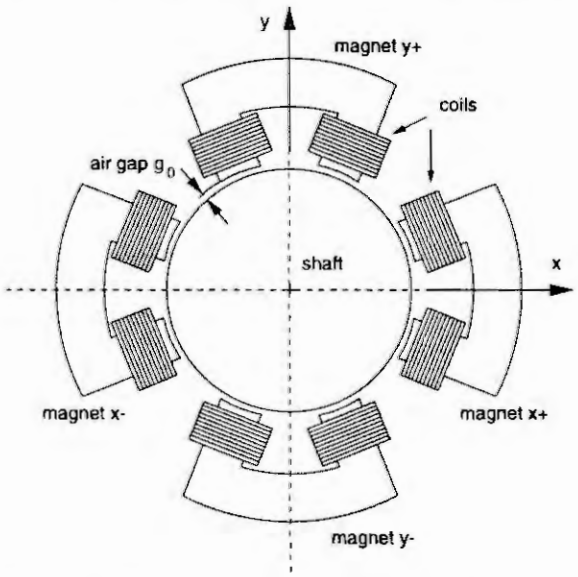


Figure 1: Schematic diagram of an active magnetic bearing with 4 actuators (electromagnets).

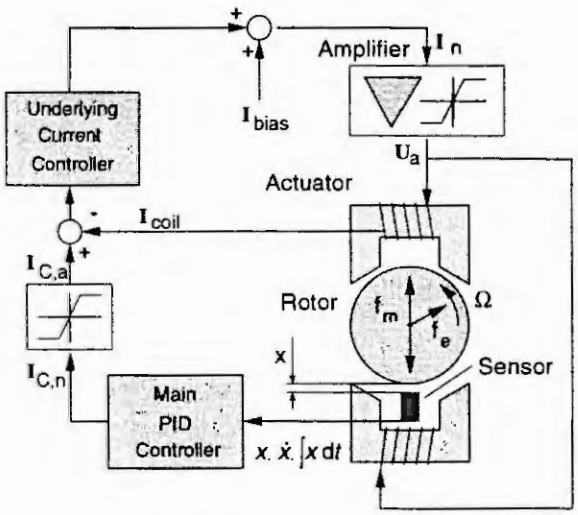


Figure 2: Schematic of the feedback control loop of one axis of the active magnetic bearing model

System equations

The mathematical formulation of the model consists of a pair of second-order differential equations, each of them describing the motion of the shaft in the $x-$ and $y-$direction, respectively. They can be represented in terms of the non-dimensional rotor displacements $X = x/g_0, Y = y/g_0$ in the form

$$\begin{aligned} X'' &= \left(\frac{2\pi}{\Omega}\right)^2 \left[\frac{1+\sum L_{mi,0}}{4\left[G_{PN}\left(1+\sum L_{mi,0}\right)-1\right]}\left(\Phi_{N,x+}^2 - \Phi_{N,x-}^2\right) + E\Omega^2\cos\left(2\pi\tau\right)\right], \\ Y'' &= \left(\frac{2\pi}{\Omega}\right)^2 \left[\frac{1+\sum L_{mi,0}}{4\left[G_{PN}\left(1+\sum L_{mi,0}\right)-1\right]}\left(\Phi_{N,y+}^2 - \Phi_{N,y-}^2\right) + E\Omega^2\sin\left(2\pi\tau\right)\right]. \end{aligned} \tag{1}$$

Additionally, a set of four first-order differential equations is used for the magnetic fluxes $\Phi_{N,x\pm}, \Phi_{N,y\pm}$ in each of the magnets x± and y± (see Fig. 1)

$$\begin{aligned} \Phi'_{N,z+} &= \frac{2\pi}{\Omega} R_N \left[\left(1+\sum L_{mi,0}\right) U_{a,z+} - \Phi_{N,z+}\left(1 - Z + \sum L_{mi,z+}\right)\right], \\ \Phi'_{N,z-} &= \frac{2\pi}{\Omega} R_N \left[\left(1+\sum L_{mi,0}\right) U_{a,z-} - \Phi_{N,z-}\left(1 + Z + \sum L_{mi,z-}\right)\right], \end{aligned} \tag{2}$$

where z, Z have to be substituted by x, X and y, Y, respectively. The symbol $(')$ denotes the derivative with respect to the dimensionless time $\tau = \dfrac{\Omega\omega_0 t}{2\pi}$, with Ω as the dimensionless rotor speed and ω_0 as the natural frequency of the linearized system at the shaft's center position. The constant E represents the non-dimensional unbalance eccentricity of the rotor, R_N is a combined material and design constant and G_{PN} is the proportional gain of the PID-controller.

The magnetic reluctance of an actuator (electromagnet) along the path of the magnetic flux is represented by $\sum L_{mi,z\pm}$ the sum of i non-dimensional reluctance values in the stator of the magnetic bearing

and in the lamination of the rotor. Each single reluctance $L_{mi,z\pm}$ is represented by a nonlinear function as

$$L_{mi,z\pm} = L_{mi,0}\frac{\mu_{r,mi,0}}{\mu_{r,mi}}, \quad \mu_{r,mi} = \frac{1-\mu_{r,mi,0}}{\pi}\text{atan}\left(K_{M,mi}\frac{|\Phi_{N,z\pm}| - \Phi_{N,mi,max}}{\Phi_{N,mi,max}}\right) + \frac{1+\mu_{r,mi,0}}{2}. \quad (3)$$

The linear range of function (3) is determined by $L_{mi,0}$. Magnetic saturation is taken into account by introducing the relative permeability $\mu_{r,mi}$ as a function of the magnetic flux Φ_N. By using the atan-function and a shape factor $K_{M,mi}$ the relative permeability $\mu_{r,mi}(\Phi_N)$ can be approximated such that it decreases smoothly to almost one for values $\Phi_N > \Phi_{N,mi,max}$ of the magnetic flux.

The actual voltage $U_{a,z\pm}$ that is used in Eqs.(2) is calculated from

$$\begin{aligned} U_{a,z\pm} &= G_{P,I}\left(I_{C,a,z\pm} - I_{z\pm}\right) + 1 + \quad (4)\\ &+ \left[\frac{1}{2} + \frac{1}{\pi}\text{atan}\left(K_U\frac{G_{P,I}\left(I_{C,a,z\pm} - I_{z,+}\right) + 1 - U_{max}}{U_{max}}\right)\right]\left[U_{max} - G_{P,I}\left(I_{C,a,z\pm} - I_{z\pm}\right) - 1\right]\\ &- \left[\frac{1}{2} - \frac{1}{\pi}\text{atan}\left(K_U\frac{G_{P,I}\left(I_{C,a,z\pm} - I_{z\pm}\right) + 1 + U_{max}}{U_{max}}\right)\right]\left[U_{max} + G_{P,I}\left(I_{C,a,z\pm} - I_{z\pm}\right) + 1\right]. \end{aligned}$$

Note that $U_{z,+,a}$ is limited to a maximum value of U_{max}. Parameter $G_{P,I}$ represents the proportional gain of the underlying current controller. The coil currents needed for this controller are given by

$$\begin{aligned} I_{C,a,z\pm} = I_{C,n,z\pm} &+ \left[\frac{1}{2} + \frac{1}{\pi}\text{atan}\left(K_I\frac{I_{C,n,z\pm} - I_{C,max}}{I_{C,max}}\right)\right]\left[I_{C,max} - I_{C,n,z\pm}\right] -\\ &- \left[\frac{1}{2} - \frac{1}{\pi}\text{atan}\left(K_I\frac{I_{C,n,z\pm}}{I_{C,max}}\right)\right] I_{C,n,z\pm} \quad (5) \end{aligned}$$

and

$$I_{z\pm} = \frac{1 \mp Z + \sum L_{mi,z\pm}}{1 + \sum L_{mi,0}}\Phi_{N,z\pm}, \qquad (z = x, y). \quad (6)$$

The nominal coil currents in Eq.(5) are calculated by the main PID-controller as

$$I_{C,n,z\pm} = 1 \mp G_{PN}Z \mp G_{DN}\frac{\Omega}{2\pi}Z' \mp G_{IN}\frac{2\pi}{\Omega}\int_0^\tau Z\,\mathrm{d}\tau \quad (7)$$

with G_{PN}, G_{DN}, and G_{IN} as the proportional, differential and integral feedback gain, respectively. A more detailed discussion of the system equations can be found in [5].

To sum up the presented model is described by a system of non-autonomous, nonlinear differential equations with a total of 10 state variables. The mathematical functions to realize magnetic saturation or to limit maximum amplifier output were carefully selected in order to keep the system equations differentiable and to avoid discontinuities. This is especially important for the chosen numerical solution methods.

Numerical solution methods

Since an analytical solution cannot be found for the presented mathematical model numerical methods have to be employed. Numerical simulation is a well-known tool and is frequently used for this kind of problems. However, the simulation approach, based on the Initial Value Problem (IVP) is rather slow, in particular close to stability limits of the system. Furthermore, unstable solutions of limit cycles cannot be calculated by numerical simulation. However, a positive aspect of the simulation approach via IVP is that it is not restricted to periodic solutions, and any kind of quasi-periodic or even chaotic solutions can be found.

In this investigation the numerical simulation method is an additional tool only, primarily used for verification and for calculating time response plots. It is obvious that the steady-state response of the system is basically periodic with the period T of one rotor revolution. Therefore, numerical integration is used and formulated as a Boundary Value Problem (BVP). By introducing periodic boundary conditions

$z(0) = z(T)$ and solving the BVP numerically any kind of stable or unstable periodic solutions can be calculated [2]. To track solutions for a certain parameter variable, which in our case is the exciting unbalance frequency Ω, a continuation (path-following) algorithm can be employed. This becomes very important if so-called turning points occur during the continuation process.

The following results were obtained using the subroutine collection BIFPACK, see [3]. BIFPACK uses a multiple shooting routine to solve boundary value problems. It has implemented a path-following algorithm and it can investigate the stability of a periodic solution. Beside other useful features the package offers a branch switching option to calculate new starting points on solutions branching off at a bifurcation.

Numerical results

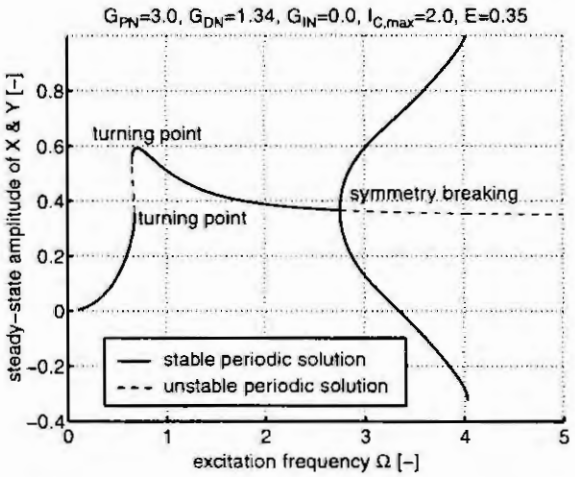


Figure 3: Response of the AMB-supported rotor due to imbalance excitation. Zero integral feedback gain. (BVP-result)

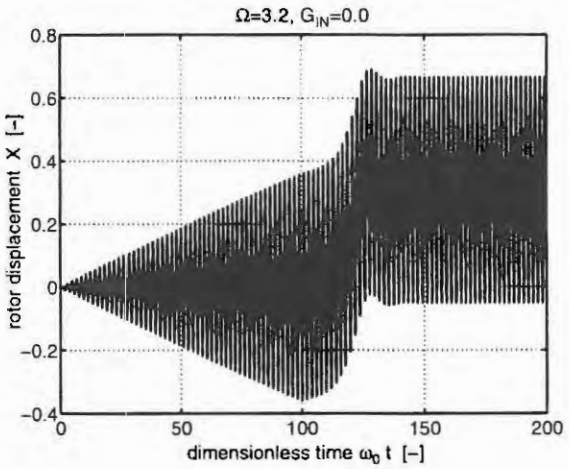


Figure 4: Time history of rotor displacement at $\Omega = 3.2$ for AMB-system without integral feedback gain.

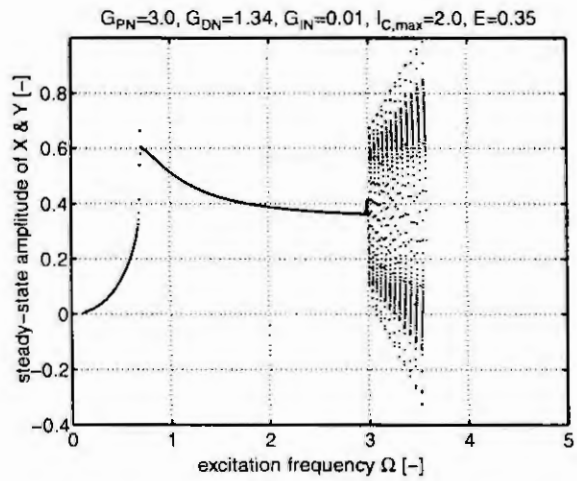


Figure 5: Response of the AMB-supported rotor due to imbalance excitation. Integral feedback gain $G_{IN} = 0.01$. (IVP-result)

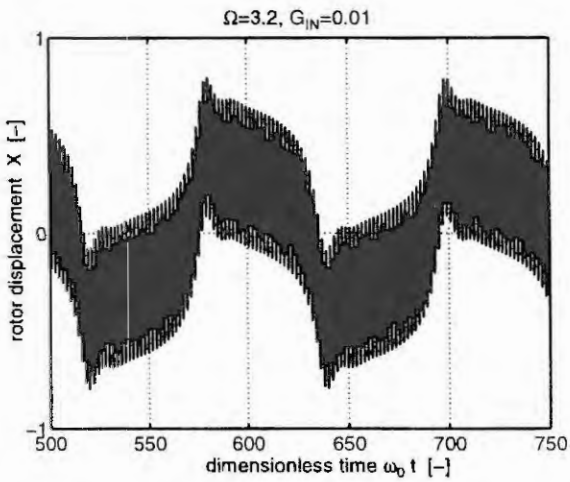


Figure 6: Time history of rotor displacement at $\Omega = 3.2$ for system with integral feedback gain $G_{IN} = 0.01$.

Solving the BVP for the system equations, bifurcation diagrams can be generated with the rotor speed Ω as the selected bifurcation parameter. Numerical simulation is used for verification and interpretation

of the obtained stable or unstable periodic solutions. Since a fully symmetrical problem is discussed, all diagrams are valid for both directions x and y.

In the following, system responses are discussed for certain parameter values in order to show a few of the most interesting results of this research. The system response without integral feedback is shown in Fig. 3. This result is obtained by the BVP-approach using BIFPACK. Stepping from low to high values of the excitation frequency, at $\Omega = 0.67$ a turning point occurs. Because of the softening spring characteristic of the system the resonant curve leans to the left until a second turning point at $\Omega = 0.64$ is reached. For frequency values between these two turning points three coexisting periodic solutions exist, i.e. two stable solutions and one unstable solution.

Increasing the frequency far beyond the eigenfrequency of the linearized system at $\Omega = 1.0$ symmetry breaking occurs at $\Omega = 2.75$. At this bifurcation point the symmetric periodic solution becomes unstable, when stable unsymmetric solutions emanates. The maximum and minimum values of the rotor vibrations are not equal in the case of a nonsymmetric solution. To verify these results, additional numerical simulations are carried out. Figure 4 shows the time history of the rotor deflections for an increasing unbalance eccentricity from zero to maximum within 100 time units at a constant excitation frequency of $\Omega = 3.2$. The nonlinear behavior of the system is clearly shown when the symmetric oscillations become unstable for large deflections and a nonsymmetric oscillation is developed.

With an integral feedback gain $G_{IN} = 0.01$ stable periodic branches could not be found anymore beyond the speed threshold of $\Omega = 2.75$. Instead, a quasiperiodic motion of the rotor occurs, which is detected by numerical simulation results, see Fig. 5. Because of symmetry breaking the integral part of the PID-controller tries to compensate the offset of the position signal. However, due to saturation within the control loop the system toggles between two (now unstable) solutions beyond the symmetry breaking bifurcation point. Figure 6 shows the corresponding time history at a constant excitation frequency of $\Omega = 3.2$. The quasi-periodic nature of this oscillation can easily be recognized.

Conclusions

An active magnetic bearing consists of several nonlinear components which have to be represented in a comprehensive mathematical model. Saturation of the magnetic flux as well as saturation of the amplifier output are among the most important nonlinearities. Such limitations of system variables have to be modelled carefully with respect to the numerical solution method and the physics the effect is based on, to obtain reliable results. The presented model leads to unexpected results which are not predictible by a linear model. However, further improvements are necessary to include some more nonlinearities as for example geometric coupling of both actuator axes.

Acknowledgement

The authors gratefully acknowledge the support of this project by the Austrian science foundation *Fonds zur Förderung der wissenschaftlichen Forschung (FWF)* and the *Institute of Machine Dynamics and Measurements* at the Vienna University of Technology, Austria.

References

[1] Schweitzer, G., Traxler, A., Bleuler, H., *Magnetlager, Grundlagen, Eigenschaften und Anwendungen berührungsfreier, elektromagnetischer Lager*, Springer Verlag, Berlin, 1993.

[2] Troger, H., Steindl, A., *Nonlinear Stability and Bifurcation Theory*, Springer: Wien - New York, 1991.

[3] Seydel R., *BifPack, a Program Package for Continuation, Bifurcation and Stability Analysis*, Program documentation, University of Ulm, Germany, 1996.

[4] Steinschaden, N., Springer, H., *Some Nonlinear Effects of Magnetic Bearings*, Proc. of the 1999 ASME Design Engineering Technical Conferences, Las Vegas, Nevada, ASME Conf. paper No. DETC99/VIB-8063, 1999.

[5] Steinschaden, N., Springer, H., *Nonlinear Stability Analysis of Active Magnetic Bearings*, Proc. of 5th Int. Symp. on Magnetic Susp. Tech., Santa Barbara, CA, Dec.1999 (to be publ. by NASA).

MODELLING AND CONTROL OF 3D OVERHEAD CRANES

A. Giua, M. Sanna, C. Seatzu
Dip. di Ingegneria Elettrica ed Elettronica, Università di Cagliari, Italy
email: {giua,seatzu}@diee.unica.it.

Abstract. In this paper we deal with the problem of designing a controller for a three–dimensional overhead crane. We consider a linear model of the crane where the length of the suspending rope is a time–varying parameter. The set of models given by frozen values of the rope length can be reduced to a single time–invariant reference model using suitable time scalings. A controller for the reference model can be designed by assigning the desired closed loop eigenvalues for the system. The time scaling relations can be used to derive a control law for the time–varying system that implements an implicit gain–scheduling.

1. Introduction

The swinging of an object suspended from an overhead crane is an undesirable result of the crane movement and serious damage could occur during the load transport. Therefore, a satisfactory control scheme is desirable in a crane design to suppress the load swing.

Several control methodologies have been proposed in the literature [1, 3, 4, 6]. However, in quite all these cases planar cranes have been considered, i.e., it has been assumed that the movement of the load lies within a plane. On the contrary, in this paper we deal with a three–dimensional overhead crane and we propose the design of an observer–controller that aims to minimize the load swinging, while moving it to the desired position as fast as possible.

We first develop a non–linear model of the overhead crane which takes into account simultaneous travel and transverse motions. Then, under appropriate simplifying assumptions (namely, small angles, constant rope velocity, force applied by the rope equal to the weight of the load and no external force acting on the load) a linear time–varying model of the crane is obtained, where the time–varying parameter is the length of the rope that sustains the load. The linearized model has order eight and its dynamic can be described as two decoupled fourth–order systems.

The controller design is realized by first considering the set of frozen models given by different constant values of the rope length. Using two suitable time scalings, one for each sub–system, all these models can be reduced to a single time–invariant reference model that does not depend on the value of the rope length. Then, the pole placement technique enables us to design a satisfactory controller for the reference model. Finally, by inverting the time–scalings, these constant feedback gains give the corresponding time–varying gains that implement an implicit gain–scheduling.

In this paper we introduce a further improvement wrt previous works [3, 6] where a gain–scheduling approach has been adopted: a double gain–scheduling has been introduced. It consists of a variation of the desired eigenvalues of the reference stationary system depending on the load mass and on the lowering/lifting movement.

An important aspect in the approach we propose has to be mentioned: the state–feedback gains are expressed in a parametrized form, as a symbolic function of the desired closed–loop dynamics (i.e., the eigenvalues of the reference closed–loop system), rope length, rope velocity, trolley and load mass. As these parameters vary, the gains need not be recomputed by reapplying the whole design procedure but can simply be obtained by function evaluation.

A final remarks, concerning stability, needs to be done. As it is well known, gain–scheduling does not guarantee the stability of the closed–loop time–varying system. However, there exist appropriate methodologies [3], based on a Lyapunov–like theorem [7], that enables us to find upper bounds on the rate of change of the varying parameter to ensure stability. In [5] it has been shown that in the applicative case examined, this approach gives sufficiently large bounds on the rope velocity to ensure stability of the time–varying system in all nominal conditions.

2. Linear time–varying model and time scaling

A three–dimensional overhead crane is constituted by a bridge and a trolley: the trolley moves on the bridge rails and contains the motor and all the other mechanisms necessary for the movement of the load; the bridge moves in the orthogonal direction thanks to appropriate wheels located on the end truck. In this paper we will consider a three–dimensional overhead crane, whose model is sketched in figure 1. The following notation is used: m_T, m_B are the mass of the trolley and that of the bridge, respectively; $m_C = m_T + m_B$ is total mass of the crane; m_L is the mass of the load; L is the length of the suspending rope; x_T, z_T denote the displacement of the trolley with respect to (wrt) a fixed coordinate system; x_L, z_L denote the displacement of the load wrt a fixed coordinate system; $x_C = (m_T x_T + m_L x_L)/(m_T + m_L)$, $z_C = (m_C z_T + m_L z_L)/(m_C + m_L)$ denote the displacement of the center of gravity of the overall system wrt a fixed coordinate system; φ is the angle between the suspending rope and the vertical; θ is the angle between the oscillation plane of the load and the XY plane, taken as positive when clockwise; $x_V = x_T - x_L = L\,sin\varphi\,cos\theta$, $z_V = z_T - z_L = L\,cos\varphi\,sin\theta$ denote the displacement of the load wrt the vertical; f_x and f_z are the control forces applied to the trolley and to the bridge, respectively; g is the gravitation constant.

If the load is heavy enough, it is possible to consider the suspending rope as a rigid rod. Under appropriate simplifying assumptions (namely, small angles, force applied by the rope equal to the weight of the load and

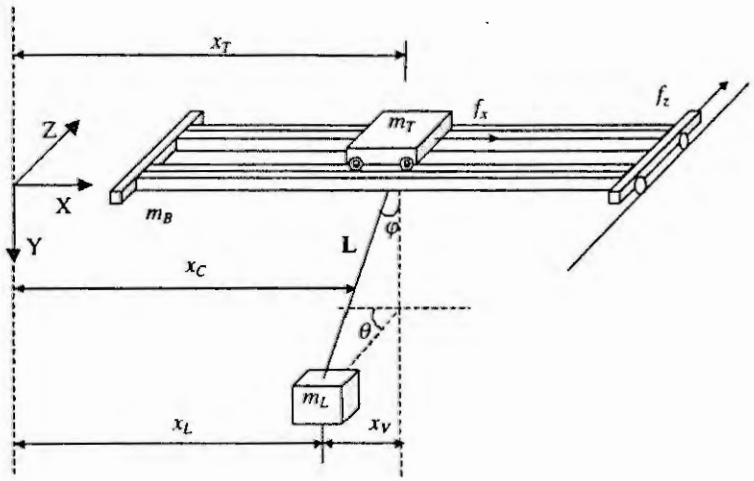


Figure 1: *Model of the 3D crane.*

no disturbance acting on the system) we obtain [5] the linearized model described by

$$\begin{cases} \ddot{x}_V + \dfrac{g(m_T + m_L)}{m_T L} x_V = \dfrac{f_x}{m_T}, \\ \ddot{x}_C = \dfrac{f_x}{m_T + m_L}, \\ \ddot{z}_V + \dfrac{g(m_C + m_L}{m_C L} z_V = \dfrac{f_z}{m_C}, \\ \ddot{z}_C = \dfrac{f_z}{m_C + m_L}. \end{cases} \tag{1}$$

Choosing the following state variables:

$$\begin{array}{llll} x_1(t) = x_V(t), & x_2(t) = x_C(t) & x_3(t) = \dot{x}_V(t), & x_4(t) = \dot{x}_C(t) \\ x_5(t) = z_V(t), & x_6(t) = z_C(t) & x_7(t) = \dot{z}_V(t), & x_8(t) = \dot{z}_C(t) \end{array} \tag{2}$$

and denoting

$$\omega_x(t) \equiv \omega_x(L(t)) = \left(\frac{g(m_T + m_L)}{m_T L(t)} \right)^{0.5}, \quad \omega_z(t) \equiv \omega_z(L(t)) = \left(\frac{g(m_C + m_L)}{m_C L(t)} \right)^{0.5}, \tag{3}$$

we get from (1) the following state variable equation:

$$\dot{\boldsymbol{x}}_t = \boldsymbol{A}_t \boldsymbol{x}_t + \boldsymbol{B}_t \boldsymbol{u}_t \tag{4}$$

with

$$\boldsymbol{x}_t = \begin{bmatrix} x_1(t) \\ \vdots \\ x_8(t) \end{bmatrix}, \quad \boldsymbol{u}_t = \begin{bmatrix} f_x(t) \\ f_z(t) \end{bmatrix},$$

$$\boldsymbol{A}_t = \left[\begin{array}{cccc|cccc} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ -\omega_x^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -\omega_z^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right], \quad \boldsymbol{B}_t = \left[\begin{array}{c|c} 0 & 0 \\ 0 & 0 \\ 1/m_T & 0 \\ 1/(m_T + m_L) & 0 \\ \hline 0 & 0 \\ 0 & 0 \\ 0 & 1/m_C \\ 0 & 1/(m_C + m_L) \end{array}\right].$$

The subscript t has been introduced to recall that the variables are functions of time. The model given by (4) is time-varying because both ω_x and ω_z are functions of $L(t)$. If we consider a given constant value of

both ω_x and ω_z, i.e., if we consider the system (4) for a frozen value of L, we can consider the following transformations:

$$\tau_x = \omega_x t, \qquad \tau_z = \omega_z t. \tag{5}$$

These transformations define a time scaling that enable us to rewrite (2) as:

$$\boldsymbol{x}_t = \boldsymbol{N}\boldsymbol{x}_\tau \tag{6}$$

where

$$\begin{aligned} \boldsymbol{N} &= \mathrm{diag}\{\ 1,\ \ 1,\ \ \omega_x,\ \ \omega_x,\ \ 1,\ \ 1,\ \ \omega_z,\ \ \omega_z\ \} \\ \boldsymbol{x}_\tau &= [\ x_1(\tau_x)\ \ x_2(\tau_x)\ \ x_3(\tau_x)\ \ x_4(\tau_x)\ \ x_5(\tau_z)\ \ x_6(\tau_z)\ \ x_7(\tau_z)\ \ x_8(\tau_z)\ ]^T. \end{aligned} \tag{7}$$

Moreover, we may also write

$$\dot{\boldsymbol{x}}_t = \boldsymbol{\Omega}\boldsymbol{N}\dot{\boldsymbol{x}}_\tau \tag{8}$$

where $\dot{\boldsymbol{x}}_\tau$ is the derivative of $\boldsymbol{x}_\tau$ wrt τ_x for the first four components and wrt τ_z for the remaining ones. It has been assumed

$$\boldsymbol{\Omega} = \mathrm{diag}\{\ \omega_x,\ \ \omega_x,\ \ \omega_x,\ \ \omega_x,\ \ \omega_z,\ \ \omega_z,\ \ \omega_z,\ \ \omega_z\ \}. \tag{9}$$

Using (6) and (8), it is possible to rewrite the equation (4) as

$$\dot{\boldsymbol{x}}_\tau = \boldsymbol{A}_\tau \boldsymbol{x}_\tau + \boldsymbol{B}_\tau \boldsymbol{u}_\tau \tag{10}$$

with

$$\boldsymbol{u}_\tau = \begin{bmatrix} \frac{1}{\omega_x^2} & 0 \\ 0 & \frac{1}{\omega_z^2} \end{bmatrix} \boldsymbol{u}_t = \boldsymbol{N}_u^{-1}\boldsymbol{u}_t = \begin{bmatrix} \frac{f_x}{\omega_x^2} \\ \frac{f_z}{\omega_z^2} \end{bmatrix}, \tag{11}$$

$$\boldsymbol{A}_\tau = \boldsymbol{N}^{-1}\boldsymbol{\Omega}^{-1}\boldsymbol{A}_t\boldsymbol{N} = \left[\begin{array}{cccc|cccc} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right], \tag{12}$$

$$\boldsymbol{B}_\tau = \boldsymbol{N}^{-1}\boldsymbol{\Omega}^{-1}\boldsymbol{B}_t\boldsymbol{N}_u = \left[\begin{array}{cccc|cccc} 0 & 0 & 1/m_T & 1/(m_T+m_L) & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 1/m_C & 1/(m_C+m_L) \end{array}\right]^T. \tag{13}$$

The representation given by (10) is time–invariant and does not depend on the frozen value of L in (4).

3. Controller design

Let us consider a linear and time–invariant system of the form (10). If the couple $(\boldsymbol{A}_\tau, \boldsymbol{B}_\tau)$ is controllable [2], then a regulator can be designed by imposing the closed loop poles to system (10), finding a control law of the form

$$\boldsymbol{u}_\tau = -\boldsymbol{K}_\tau \boldsymbol{x}_\tau \tag{14}$$

where $\boldsymbol{K}_\tau$ is a constant matrix and does not depend on the value of L. The above equation can be transformed, using (6) and (11), into a corresponding law for the frozen system (4) that gives:

$$\boldsymbol{u}_t = -\boldsymbol{K}_t \boldsymbol{x}_t, \qquad \boldsymbol{K}_t = \boldsymbol{N}_u \boldsymbol{K}_\tau \boldsymbol{N}^{-1}. \tag{15}$$

The feedback laws (14) and (15) lead to closed loop systems whose characteristic matrices are:

$$\bar{\boldsymbol{A}}_\tau = \boldsymbol{A}_\tau - \boldsymbol{B}_\tau \boldsymbol{K}_\tau, \qquad \bar{\boldsymbol{A}}_t = \boldsymbol{A}_t - \boldsymbol{B}_t \boldsymbol{K}_t. \tag{16}$$

Note that also the above matrices can be rewritten as

$$\bar{\boldsymbol{A}}_t = \begin{bmatrix} \bar{\boldsymbol{A}}_{t,x} & \boldsymbol{0}_{4,4} \\ \boldsymbol{0}_{4,4} & \bar{\boldsymbol{A}}_{t,z} \end{bmatrix} \qquad \bar{\boldsymbol{A}}_\tau = \begin{bmatrix} \bar{\boldsymbol{A}}_{\tau,x} & \boldsymbol{0}_{4,4} \\ \boldsymbol{0}_{4,4} & \bar{\boldsymbol{A}}_{\tau,z} \end{bmatrix}.$$

For a stationary system it is easy to find a feedback control law by imposing the closed loop eigenvalues following the procedure presented in [2]. Let us denote as $s^4 + a_{x,3}s^3 + a_{x,2}s^2 + a_{x,1}s + a_{x,0}$ and $s^4 + a_{z,3}s^3 + a_{z,2}s^2 + a_{z,1}s + a_{z,0}$ the open loop characteristic polynomials relative to matrices $A_{\tau,x}$ and $A_{\tau,z}$, respectively. Then, let $s^4 + p_{x,3}s^3 + p_{x,2}s^2 + p_{x,1}s + p_{x,0}$ and $s^4 + p_{z,3}s^3 + p_{z,2}s^2 + p_{z,1}s + p_{z,0}$ be the desired closed loop

characteristic polynomials relative to matrices $\bar{A}_{\tau,x}$ and $\bar{A}_{\tau,z}$, respectively. Therefore, the time–invariant control law is [2]:

$$K_\tau = \begin{bmatrix} K_{\tau,x} & 0_{1,4} \\ 0_{1,4} & K_{\tau,z} \end{bmatrix} P_c^{-1} \tag{17}$$

where

$$K_{\tau,x} = \begin{bmatrix} p_{x,0} - a_{x,0} & p_{x,1} - a_{x,1} & p_{x,2} - a_{x,2} & p_{x,3} - a_{x,3} \end{bmatrix},$$

$$K_{\tau,z} = \begin{bmatrix} p_{z,0} - a_{z,0} & p_{z,1} - a_{z,1} & p_{z,2} - a_{z,2} & p_{z,3} - a_{z,3} \end{bmatrix},$$

$$P_c = \begin{bmatrix} (A_{\tau,x}^3 + a_{x,3}A_{\tau,x}^2 + a_{x,2}A_{\tau,x} + a_{x,1}I)B_{\tau,x} \\ (A_{\tau,x}^2 + a_{x,3}A_{\tau,x} + a_{x,2}I)B_{\tau,x} \\ (A_{\tau,x} + a_{x,3}I)B_{\tau,x} \\ B_{\tau,x} \\ (A_{\tau,z}^3 + a_{z,3}A_{\tau,z}^2 + a_{z,2}A_{\tau,z} + a_{z,1}I)B_{\tau,z} \\ (A_{\tau,z}^2 + a_{z,3}A_{\tau,z} + a_{z,2}I)B_{\tau,z} \\ (A_{\tau,z} + a_{z,3}I)B_{\tau,z} \\ B_{\tau,z} \end{bmatrix}^T,$$

and $B_{\tau,x}$ and $B_{\tau,z}$ are the two non–null sub–matrices of B_τ, i.e.,

$$B_\tau = \begin{bmatrix} B_{\tau,x} & 0_{4,1} \\ 0_{4,1} & B_{\tau,z} \end{bmatrix}.$$

Note that, P_c is an equivalence transformation that brings the initial system into a controllable canonical form [2].

Using equation (15), we get the time–varying control law:

$$K_t = \begin{bmatrix} K_{t,x} & 0_{1,4} \\ 0_{1,4} & K_{t,z} \end{bmatrix} \tag{18}$$

where

$$K_{t,x} = \begin{bmatrix} (p_{x,2} - p_{x,0} - 1)m_T\omega_x^2 & p_{x,0}(m_T + m_L)\omega_x^2 & p_{x,3} - p_{x,1})m_T\omega_x & p_{x,1}(m_T + m_L)\omega_x \end{bmatrix}$$

$$K_{t,z} = \begin{bmatrix} (p_{z,2} - p_{z,0} - 1)m_C\omega_z^2 & p_{z,0}(m_C + m_L)\omega_z^2 & p_{z,3} - p_{z,1})m_C\omega_z & p_{z,1}(m_C + m_L)\omega_z \end{bmatrix}.$$

4. An applicative example

In this section we show how the above procedure can be applied to a real overhead crane. We consider a model produced by Munck Cranes Inc., Ontario–Canada whose load capacity varies from 1 to 50 ton. In particular, in this paper we consider an overhead crane whose trolley mass is $m_T = 4037\,\mathrm{Kg}$ and whose bridge mass is $m_B = 4112\,\mathrm{Kg}$. We assume the length of the suspending rope to be: $L(t) \in [L_{min}, L_{max}]$, where $L_{min} = 2\,\mathrm{m}$ and $L_{max} = 10\,\mathrm{m}$. To deduce the controller and observer gain matrices we assumed that the rope length has a constant derivative $|\dot{L}(t)| = 0.5\,\mathrm{m/s}$. Clearly this is not true during a real movement. Therefore during numerical simulations, we have removed this assumption and we have imposed an acceleration of $\pm 0.5\,\mathrm{m/s^2}$ at the beginning and at the end of the hoisting and lowering movement, while in the central part of the movement the velocity is constant and equal to $\pm 0.5\,\mathrm{m/s}$.

During the simulations, we have also removed the assumption of linearity thus we used a nonlinear model of the crane derived in [5]. Numerical simulations have been carried out with the SIMULINK toolbox of MATLAB.

An important remarks needs to be done. The physical realization of such a gain–scheduling controller requires the knowledge of all state variables (centre of mass position and velocity, load displacemeut with respect to (wrt) the vertical and its rate of change), of the rope length and of the load weight. During numerical simulations we assume that only the trolley position and the rope length can be measured by appropriate sensors as discussed by several authors [8] and we also design a time–varying observer via gain–scheduling and pole–placement to provide an estimate of the unknown state vector. The design procedure adopted is the same as that already presented for the observer, with the only difference that in this case, desired poles are assigned to the closed–loop stationary error system that is defined by means of the same time–scaling relations used for the controller design. Details are not reported here for brevity's requirements. An interested read can look at [5] for a precise description of the problem.

Note that in previous works the authors used the gain–scheduling technique to derive a satisfactory control law for a given planar crane [3, 6]. In those works, even in the second one where also an observer has been designed, a single set of eigenvalues for the controller and a single one for the observer has been used. In this paper, we make a different choice motivated by the greater complexity of the system at hand. In particular, we divided the whole range of possible values of the load mass in three different intervals and we further distinguished among lowering and lifting movement. Then, we associated to each range a different set of eigenvalues for the reference stationary system and the error system. In this way we introduced a double gain–scheduling, thus producing a significant improvement in the performance of the controlled

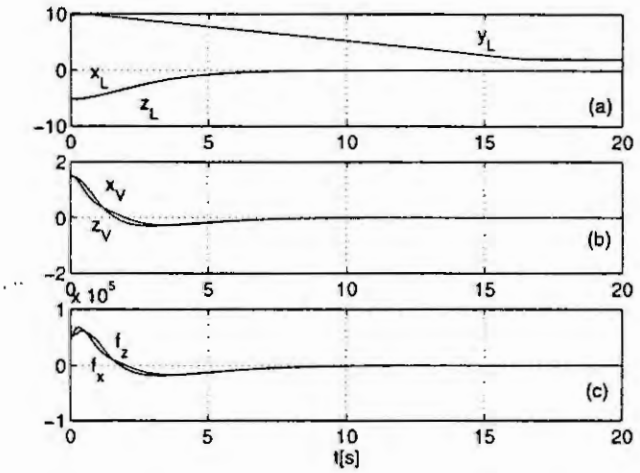


Figure 2: *The results of numerical simulation.*

system. Note that, from an applicative point of view, this does not introduce any amount in the cost of realization of the system, being the load mass assumed known [8] during each operation. These values are not reported here for brevity's sake, but are available in [5].

Now, let us present the results of a numerical simulation. We considered a load mass equal to the maximum load capacity, i.e., equal to 50 ton. The simulation was performed for a lifting movement from $L_o = 10\,\mathrm{m}$ to $L_f = 2\,\mathrm{m}$. The initial state of the crane was $x_V(0) = z_V(0) = 1.5\,\mathrm{m}$, $x_C(0) = z_C(0) = -5\,\mathrm{m}$, $\dot{x}_V(0) = \dot{x}_C(0) = \dot{z}_V(0) = \dot{z}_C(0) = 0\,\mathrm{m/s}$, while the initial state of the observer was $\hat{x}_V(0) = 1\,\mathrm{m}$, $\hat{x}_C(0) = -4.5\,\mathrm{m}$, $\hat{z}_V(0) = 2\,\mathrm{m}$, $\hat{z}_C(0) = -5.5\,\mathrm{m}$, $\hat{\dot{x}}_V(0) = \hat{\dot{x}}_C(0) = \hat{\dot{z}}_V(0) = \hat{\dot{z}}_C(0) = 0\,\mathrm{m/s}$.

In figure 2 the results of this simulation are reported. Figure (a) shows the displacement of the load wrt to a fixed coordinate system; (b) shows the displacement of the load wrt the vertical and enables us to conclude that quite no oscillation occurs during the load movement; in (c) the curves representative of the coutrol forces are shown.

5. Conclusions

In this paper we presented a general methodology for controlling three–dimensional overhead cranes. This work is an extension of previous ones where the authors limit to consider planar cranes.

Time scaling relations have been used to reduce the original time–varying system to a stationary one. The controller design for the reference system has been carried out via pole–placement. Then, the time–scalings inversion enabled us to derive in a parametric form the time–varying gains for the controller. Note that in this paper we implemented a double gain–scheduling, being the eigenvalues of the closed–loop system dependent on the load mass and on the lowering/lifting movement.

References

[1] J.W. Auernig, H. Troger, *Time optimal control of overhead crane with hoisting of the load*, Automatica, Vol. 23, N. 4, pp. 437–447, 1987.

[2] C.T. Chen, *Linear System Theory and Design*, Holt, Rinehart and Winston, Inc., 1984.

[3] G. Corriga, A. Giua, G. Usai, *An implicit gain–scheduling controller for cranes*, IEEE Trans. on Control Systems Technology, Vol. 6, N. 1, pp. 15–20, 1998.

[4] Y. Sakawa, Y. Shindo, *Optimal control of container cranes*, Automatica, Vol. 18, N. 3, pp. 257–266, 1982.

[5] M. Sanna, *Observer–controller design for three degrees of freedom cranes via gain–scheduling*, Laurea Thesis, University of Cagliari, DIEE (in italian), 1999.

[6] C. Seatzu, A. Giua, *Observer–Controller design for cranes via pole placement and gain–scheduling*, 6th IEEE Med. Conf. on Control and Automation, Alghero, Italy, June 1998.

[7] J.S. Shamma, *Analysis and design of gain–scheduled control systems*, Doctoral Thesis, Lab. Information and Decision Systems, Massachussetts Ins. of Tech. Cambridge, Massachussetts, May 1988.

[8] J. Virkkunen, A. Marttinen, K. Rintanen, R. Salminen, J. Seitsonen, *Computer control of over–head and gantry cranes*, Proc. IFAC 11th World Congress, Tallin, Estonia, pp. 401–405, 1990.

MODELLING AND SIMULATION OF A GRIPPER

G. Ferretti, C. Maffezzoni, G. Magnani and P. Rocco
Politecnico di Milano, Dipartimento di Elettronica e Informazione
Piazza Leonardo da Vinci 32, 20133, Milano, Italy

Abstract. A modular approach to the modelling and simulation of a gripper for space robotics applications is discussed in this paper. The key features allowing modularity are a new model of the point contact and the adoption of the modelling and simulation environment MOSES [5], based on the concepts of Object-Oriented Modelling.

Introduction

In this paper, a modular approach to the modelling and simulation of a gripper, designed for space robotics applications, is presented. In particular, the attention is focused on the model of tendons and on a new model of point contact, based on an extension of the LUGRE model of friction [3]. The modular approach is indeed allowed by the new model in that it is *local* to the point of contact, i.e. it does not consider the grasping system as a whole, as in the case of the classical approach, and it is applicable to an arbitrary number of contacts.

The model of the gripper has been built by aggregation of several submodels in the MOSES environment [5], a general purpose environment developed at the Department of Electronics and Information of the Politecnico di Milano. MOSES implements the basic concepts of Object-Oriented Modeling [7] such as: 1) declarative definitions of models; 2) standardization of the interfaces among (sub)models; 3) Object-Oriented design and data management. Another specific feature of MOSES is the symbolic manipulation step, described in detail in [6], which generates an efficient DAE (Differential Algebraic Equations) system through symbolic manipulation of the raw assembly of submodel equations, to be solved by the numerical solver.

Description of the gripper

The gripper considered in this work (described in more detail in [8]) has been mainly designed for space robotic manipulation, to achieve large workspace and high grasping versatility in micro gravity conditions. It is made up by three independently actuated fingers (Fig. 1), moving radially with respect to the wrist center along lines spaced by 120°. In turn, each finger is made up by three phalanxes (Q, B and F in Fig. 2), connected by tendons T1, T2 and T3 and by pulleys P1, P2, P3, P4 and P5. Pulleys P1 and P3 are fixed to the wrist and the radius of P1 is twice that of P3, pulleys P2 and P5 are fixed to phalanx B and F respectively, while pulley P4 is idle.

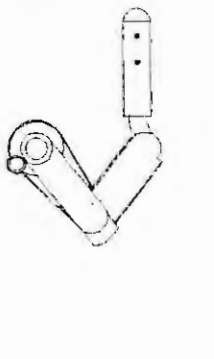


Fig. 1 One finger of the gripper

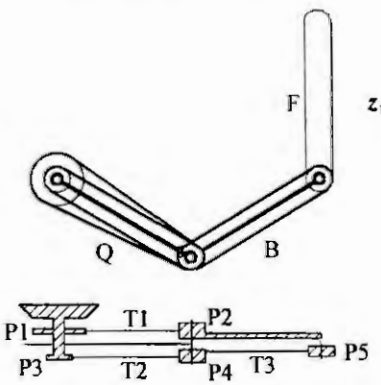


Fig. 2 Tendon linkage

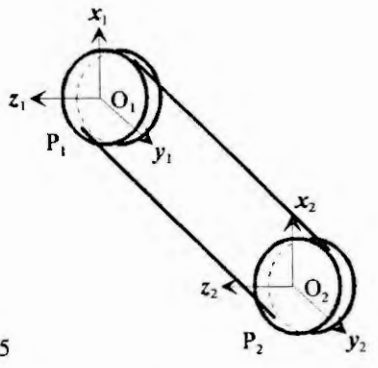


Fig. 3 Tendon terminal frames

Each finger is therefore a serial chain of three links connected by rotary joints and additionally constrained by the tendons, which maintain the last phalanx parallel to the wrist (approach) axis while moving; the moving torque is applied to phalanx Q. Since the dynamics of the serial chain has been described through the well known Newton-Euler approach, widely considered in the literature, attention will be focused on the tendon model and on a new model of the point contact.

The tendon model

In a modular approach, models should be written in terms of the variables associated to the *terminals*, conceived as the only mean to establish an interaction among models. In particular, each *mechanical* terminal has associated a reference frame i, with O_i being the origin and $\mathbf{x}_i, \mathbf{y}_i, \mathbf{z}_i$ being the unit vectors, and the following terminal variables: $\mathbf{p}_i$ = vector position from the origin of the absolute frame to O_i; $\mathbf{v}_i$ = absolute linear velocity of O_i; $\mathbf{a}_i$ = absolute linear acceleration of O_i; $\mathbf{R}_i$ = rotation matrix between frame i and the absolute frame; $\mathbf{w}_i$ = absolute angular velocity of frame i; $\boldsymbol{\varepsilon}_i$ = absolute angular acceleration of frame i; $\mathbf{f}_i$ = force acting at O_i; $\mathbf{n}_i$ = torque acting at O_i. All vector quantities are defined with respect to frame i. In the case of the tendon model one mechanical terminal is associated to each pulley P1 and P2 in Fig. 3, with the origin O_i in the center of the pulley and with the $\mathbf{z}_i$ axis along the axis of rotation of the pulley.

Since the elastic behavior of the tendons may be important, particularly in determining the grasping force dynamics, a model accounting explicitly for the tendon elongation has been developed. If an elastic tendon is assumed, the torque component along the rotation axis applied to one of the two pulleys, say P2, can be defined as a known (linear) function of the elongation e and of its time derivative e':

$$\mathbf{n}_2 \cdot \mathbf{z} = f(e, e') = K_t e + D_t e'$$

with $\mathbf{z} = [\ 0 \quad 0 \quad 1\ ]^T$ and e, e' given by:

$$\begin{aligned} \omega_i &= \mathbf{w}_i \cdot \mathbf{z} - \frac{(\mathbf{R}_2\mathbf{v}_2 - \mathbf{R}_1\mathbf{v}_1) \cdot [\mathbf{R}_1\mathbf{z} \times (\mathbf{R}_2\mathbf{p}_2 - \mathbf{R}_1\mathbf{p}_1)]}{\|\mathbf{R}_2\mathbf{p}_2 - \mathbf{R}_1\mathbf{p}_1\|^2} \qquad i = 1, 2 \\ \frac{de}{dt} &= e' = r_1\omega_1 - r_2\omega_2 \end{aligned}$$

where r_1, r_2 are the radii of the pulleys and ω_1, ω_2 are the relative angular velocities of the pulleys with respect to their support. The forces $\mathbf{f}_1$ and $\mathbf{f}_2$, exerted by the tendon on the pulleys are internal forces and do not have any influence on motion or grasping, therefore it may be set: $\mathbf{f}_1 = \mathbf{f}_2 = \mathbf{0}$. The same holds for the torque components normal to the rotation axis of the pulleys, while the balance of the components along the said axis gives:

$$\frac{\mathbf{n}_1 \cdot \mathbf{z}}{r_1} + \frac{\mathbf{n}_2 \cdot \mathbf{z}}{r_2} = 0$$

The point contact model

The contact force exerted by the fingers on the object surface is resolved into a compressive *normal* force, acting along the common normal to the contacting surfaces, and a *tangential* force, attributed to friction. Moreover, if a compliant contact is assumed, an elastic strain σ may be defined, together with a relative velocity between the surfaces, $\mathbf{v}_r = [\ v_{rx} \quad v_{ry} \quad v_{rz}\ ]^T$, v_{rz} being the component along the common normal and v_{rx}, v_{ry} being the components of the relative sliding velocity.

In order to model the energy dissipation at the contact, the normal force f_{cz} cannot be defined as a function of the strain σ only, it is necessary to consider also the strain rate v_{rz}. Moreover, the energy dissipation appears to be correctly described by assuming a different force-strain characteristics during compression and release, defining a higher normal force during compression and a lower one during release, thus introducing *hysteresis*. The following model has been therefore assumed for the normal force:

$$f_{cz} = \left[\frac{K_n^{\max} + K_n^{\min}}{2} + \left(\frac{K_n^{\max} - K_n^{\min}}{2}\right)\operatorname{sign}(v_{rz})\right]\sigma$$

defining a linear characteristics during both compression and release but with different spring constants, $K_n^{\max}$ and $K_n^{\min}$, respectively.

The model of the tangential friction force adopted here is actually a 3D extension of the LUGRE model [3], slightly reformulated to explicitly account for the normal force f_{cz}:

$$v = \sqrt{v_{rx}^2 + v_{ry}^2}\ ; \quad z = \sqrt{z_x^2 + z_y^2}\ ; \quad g(v) = \mu_c + (\mu_s - \mu_c)\, e^{-\left(\frac{v}{v_s}\right)^2} \tag{1}$$

$$g(v)\dot{z}_x = v_{rx}\left[g(v) - \operatorname{sign}(z_x)\operatorname{sign}(v_{rx})\sigma_0 z\right]\ ; \quad g(v)\dot{z}_y = v_{ry}\left[g(v) - \operatorname{sign}(z_y)\operatorname{sign}(v_{ry})\sigma_0 z\right] \tag{2}$$

$$f_{cx} = f_{cz}\left(\sigma_0 z_x + \sigma_1 \dot{z}_x + \sigma_2 v_{rx}\right) \; ; \quad f_{cy} = f_{cz}\left(\sigma_0 z_y + \sigma_1 \dot{z}_y + \sigma_2 v_{ry}\right) \tag{3}$$

where f_{cx}, f_{cy} are the components of the tangential friction force and z may be interpreted as the average deflection of elastic *bristles*, deflecting under the action of a tangential force and generating the friction force. The parameters μ_s and μ_c are the static and dynamic friction coefficient respectively, while the viscous friction term has been assumed proportional to both sliding velocity and normal force, through the parameter σ_2.

The LUGRE model was proved to be effective in modeling some effects not accounted for by the classical discontinuous model, such as the rising of presliding displacements during stiction, the frictional lag, the varying break-away force and the Stribeck effect [1]. On the other hand, it was shown in [2] and [4] that the classical Coulomb friction model can be recovered from (1-3) as an asymptotic behavior, for high values of the parameters σ_0 and σ_1.

However, the main advantage of the model is the fact that it is *local* to the point of contact. The friction force depends only on the relative sliding velocity and not, as in the case of the classical model, on the whole system of forces acting ou the contacting bodies. This property saves modularity, allows distributing the model of grasping over the contact points and, above all, avoid hyperstatic redundancy in the case of more than three point of contacts. Model (1-3) applies in fact to an arbitrary number of contact points between fingers and grasped objects.

The modular model of the gripper

Figure 4 shows the MOSES [5] scheme of the *aggregate* model of one finger, where the chain structure is clearly recognizable. Note that three mechanical terminals are available for connecting the finger to the wrist submodel (a) and for representing the interaction of the finger with the grasped object submodel, respectively (b,c).

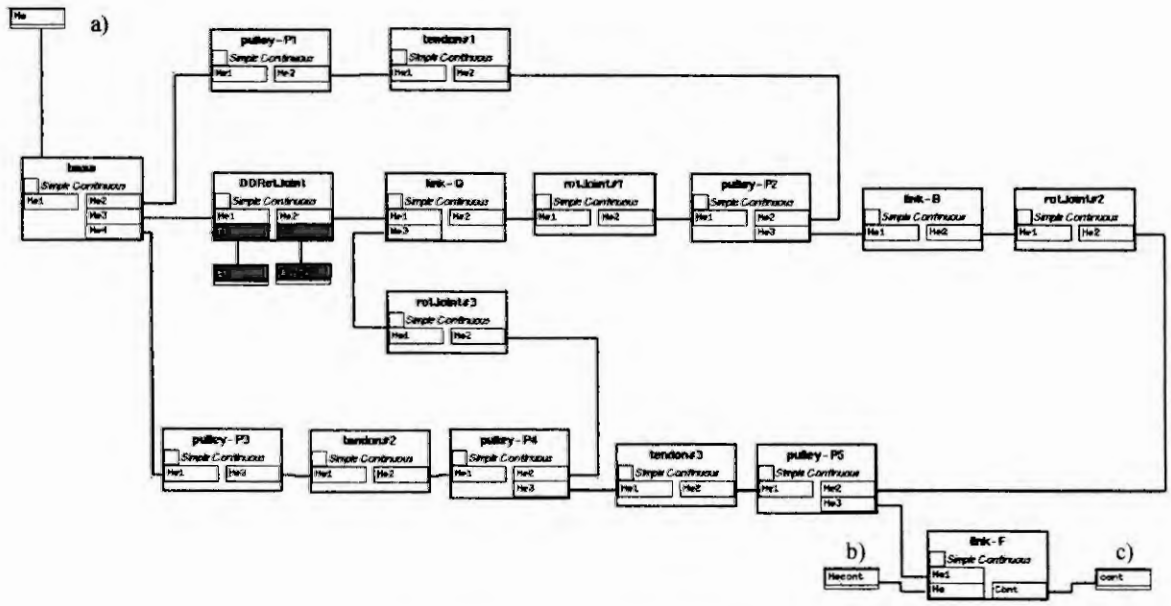


Fig. 4 Aggregate model of one finger

These external connections are established, for each finger, in the whole system model at an outer level, built by the aggregation of several submodels, including the wrist, the grasped object, the modules implementing the contact model, the finger position control (controlling the approach phase of each finger with the object to be grasped), and the force control (controlling the grasping force).

To point out the importance of the symbolic manipulation, consider that the number of differential algebraic equations of the original model (obtained by crude assembly of submodules) was more than 3000, reduced to 90 differential equations and 900 assignements after the symbolic manipulation process.

The following grasping task has been considered as an example : 1) the gripper moves initially toward a sphere, leaned on a horizontal plane and subject to gravity; 2) the gripper grasps the sphere while controlling the total grasping force; 3) the gripper lifts the sphere; 4) the grasping force set point is

decreased, the contact force on the three fingers exits from the friction cone and the sphere falls down, bouncing on the plane.

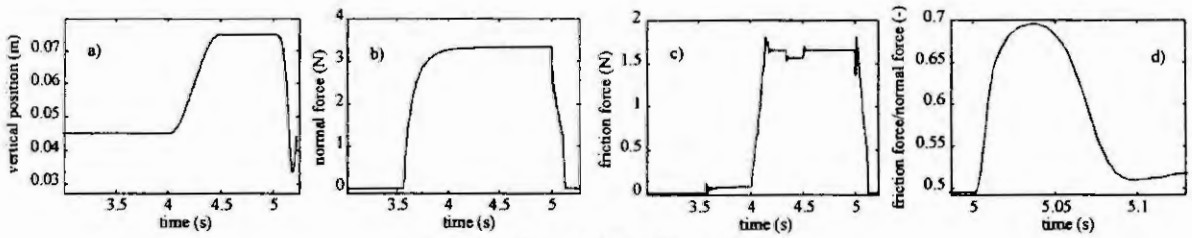


Fig. 5 Sphere grasping: simulation results

A quite soft ground is assumed, which explains the initial penetration of the sphere, having a radius of 5 cm and a mass of 0.5 kg, and the deep first bounce after release (see Fig. 5.a, where the vertical position of the center of the sphere is plotted). Since the sphere is centered with respect to the gripper's workspace and the fingers move with the same approach velocities, the contacts hold simultaneously, and the grasping force is the same for each finger: Fig. 5.b and Fig. 5.c show the normal and the tangential force amplitudes respectively. The contacts are established nearly at $t = 3.6\,\mathrm{s}$, then 0.4 s are left to achieve the force set point $\bar{f}_s = 10\,\mathrm{N}$. At $t = 4\,\mathrm{s}$ the vertical motion starts while at $t = 5\,\mathrm{s}$ the force set point is abruptly decreased to $\bar{f}_s = 1\,\mathrm{N}$. Consider now the release phase: initially the normal force decreases, while the tangential force keeps being nearly constant. The ratio friction force/normal force increases to 0.7, which is the value assumed for μ_s. At about $t = 5.04\,\mathrm{s}$ the contact force is going to exit from the friction cone, and the sphere starts sliding from the fingers. As soon as the sliding velocity becomes appreciable the ratio decreases to 0.5, the value assumed for μ_c (Fig. 5.d)).

Conclusions

A modular approach to the modelling and simulation of a gripper has been presented in this paper. Modularity is achieved through a new local model of point contact, applicable to an arbitrary number of contacts, and through the adoption of the modelling and simulation environment MOSES. Based on the model, a grasp control strategy has been designed and tested, and some simulation results have been reported.

References

1. Armstrong-Hélouvry, B., *Control of machines with friction.* Kluwer Academic Publishers, (1991).
2. Bonsignore A., Ferretti, G. and Magnani, G., Analytical formulation of the classical friction model for motion analysis and simulation. *Mathematical & Computer Modelling of Dynamical Systems*, 5, 1 (1999), pp. 43-54.
3. Canudas de Wit, C., Olsson, H., Åström, K. J. and Lischinsky, P., A new model for control of systems with friction. *IEEE Transactions on Automatic Control*, 40, 3 (1995), pp. 419-425.
4. Ferretti G., Magnani, G. and Rocco, P. Modular dynamic modeling and simulation of grasping. *IEEE/ASME International Conference on Advanced Intelligent Mechatronics AIM'99*, Atlanta, September 19-22, (1999), pp. 428-433.
5. Maffezzoni, C. and R. Girelli, MOSES: modular modeling of physical systems in an object-oriented database, *Mathematical Modeling of Systems*, 4, 2 (1998), pp.121-147.
6. Maffezzoni, C., Girelli, R. and Lluka, P., Generating efficient computational procedures from declarative models, *Simulation Practice and Theory*, 4, (1996), pp. 303-317.
7. Mattsson, S.E., Andersson, M. and Åström, K., Object Oriented modeling and simulation, in: Linkens D.A., editor, *CAD for Control Systems*, Marcel Dekker, Inc, 1993.
8. C. Melchiorri and G. Vassura, Design of a three-contact, three degree-of-freedom gripper for intravehicular robotic manipulation, *1st IFAC workshop on Space Robotics, SPRO98*, Montreal, Canada, Oct. 19-22, 1998.

Verification of Physical Parameters in a Rigid Manipulator Wrist Model

S. Hanssen[1], G.E. Hovland[2] and T. Brogårdh[3]

[1]ABB Robotics, S-721 68 Västerås, Sweden, Email: sven.hansen@se.abb.com
[2]ABB Corp. Research, N-1375 Billingstad, Norway, E-mail: geir.hovland@no.abb.com
[3]ABB Robotics, S-721 68 Västerås, Sweden, Email: torgny.brogardh@se.abb.com

Abstract

In this paper we present an identification algorithm for the rigid-body parameters of an industrial ABB robot wrist. The main contributions of our work are 1) the demonstration of rigid-body dynamic identification algorithms on large industrial robots, for which we handle complex internal couplings of the manipulator wrist, as shown in the experiments. 2) the use of the particular non-linear structure of the wrist model to solve a larger number of physical parameters than what is possible from general linear algorithms presented in the literature, given the measurement constraints on industrial robots.

Notation

Lowercase character (n, a, b, c), (q, τ), (v, β)	triads, scalar and vectorial matrices
Boldface uppercase character $(\mathbf{I})$	dyad
Uppercase character (N, A, B, C), (M)	reference frames, dyadic matrix
Boldface lowercase character $(\mathbf{r}, \mathbf{v})$, $(\mathbf{n}_i, \mathbf{a}_i, \mathbf{b}_i, \mathbf{c}_i)$	vectors, mutually perpendicular unit vectors
q_i, $\dot{q}_i$, u_i	generalised coordinates, velocities, speeds
${}^{i}m$	mass of rigid body i
${}^{k}J_{ij}$	components of moment of inertia dyads, body k
${}^{j}l_i$	length, direction i, body j
G	linear transformation matrix, motor to link

1 Introduction

The demands for path-tracking accuracy of modern industrial robots are extremely high. Current applications such as spray painting, waterjet cutting, laser cutting, plasma cutting and assembly typically demand maximum path errors of only a few tenths of a millimeter while the manipulators carry loads from less than a kilogram up to several hundred kilograms. To achieve the path tracking performance required by such applications, an extremely accurate model of the robot dynamics and the payloads of the manipulator are vital. In this paper we demonstrate a method for identifying and verifying the dynamic robot model of an industrial manipulator wrist including payload parameters.

The identification of rigid body robot dynamics has received a significant amount of attention in the literature, see for example [2, 5, 9, 11]. For most of these papers, only a subset of the physical parameters, such as mass centres, arm inertias, etc., can be identified. The remaining unknown parameters appear as linear or non-linear combinations in the identification algorithms, see section 2. Atkeson, An and Hollerback [2] describe one identification concept where 6-dof force/torque sensors are located at every single joint of the manipulator. For this case it is possible to identify all the physical parameters. However, the mounting of a 6-dof force sensor at every single joint is cumbersome and highly undesirable from a practical viewpoint.

The main contributions of our work are 1) the demonstration of rigid-body dynamic identification algorithms on large industrial robots, for which we also handle complex internal couplings of the manipulator wrist, as shown in the experiments. 2) the use of the particular non-linear structure of the wrist model to solve a larger number of physical parameters than what is possible from general linear algorithms presented in [2, 11]. The wrist model contains 25 physical parameters. From these, 18 physical

parameters can be identified directly. The remaining 7 parameters can be verified through linear and non-linear combinations. In our work these 7 parameters are verified against mechanical data obtained through CAD modelling at ABB Robotics. Mechanical data can also be obtained by measuring the individual parts of the wrist before assembly. However, some of the parameters will change after assembly due to the addition of oil and bearings. Hence, verification of the model after assembly of all parts is highly desirable.

2 Analytic Wrist Model

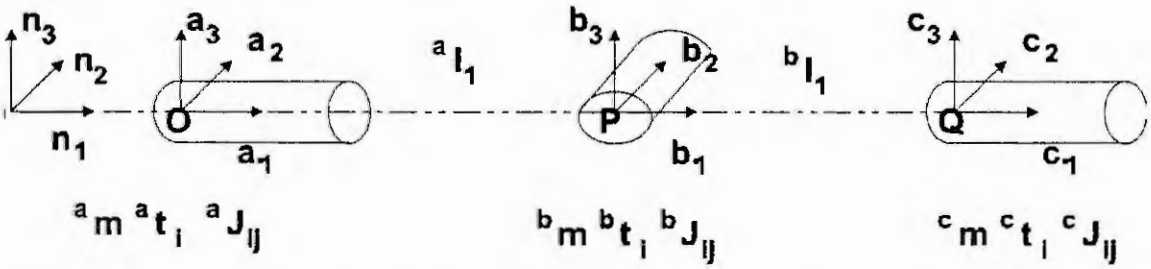


Figure 1: *Specification of wrist coordinate system and physical parameters for the 3 wrist axes, 1-3. The wrist model contains* 32 *physical parameters, of which* 25 *appear in the dynamic equations of motion.*

In this section we derive a 3-dof wrist model in which no internal couplings are considered. Then we add internal couplings, and this model we use for identification after transforming the motor variables to the arm side. The transformations of motor variables are described in section 5. The derivation of equations of motion was done in Sophia, developed by Lesser[8], with Kane's method [6]. Sophia is a set of routines to derive kinematics and dynamics in Maple. In section 2.1 we describe the kinematics of an ABB industrial robot wrist. In section 2.2 the analytical equations of motion are derived.

2.1 Kinematics

We introduce $n = \begin{pmatrix} \mathbf{n}_1 & \mathbf{n}_2 & \mathbf{n}_3 \end{pmatrix}$ as a reference triad and an origin $(\mathbf{O})$ for the inertial reference frame N. The configuration relative to the reference frame N can be written in terms of a set of independent *generalised coordinates* $q = \begin{pmatrix} q_1 & q_2 & q_3 \end{pmatrix}^T$. We introduce the reference triads, see Figure 1, as

- a rotated relative n an amount q_1 about the common $\mathbf{n}_1$-direction, a is attached to rigid body a,
- b rotated relative a an amount q_2 about the common $\mathbf{a}_2$-direction, b is attached to rigid body b,
- c rotated relative b an amount q_3 about the common $\mathbf{b}_1$-direction, c is attached to rigid body c.

The position vectors relative N of the contact points between the rigid bodies are

$$\mathbf{r}^{OP} = {}^a l_1 \mathbf{a}_1, \ \mathbf{r}^{OQ} = \mathbf{r}^{OP} + {}^b l_1 \mathbf{b}_1$$

where O, P and Q are the points shown in Figure 1. The position vectors relative N of the centre of mass for the rigid bodies are

$$\mathbf{r}^{OCMa} = a \begin{pmatrix} {}^a t_1 & {}^a t_2 & {}^a t_3 \end{pmatrix}^T, \ \mathbf{r}^{OCMb} = \mathbf{r}^{OP} + b \begin{pmatrix} {}^b t_1 & {}^b t_2 & {}^b t_3 \end{pmatrix}^T, \ \mathbf{r}^{OCMc} = \mathbf{r}^{OQ} + c \begin{pmatrix} {}^c t_1 & {}^c t_2 & {}^c t_3 \end{pmatrix}^T$$

The rigid body velocities for each body's centre of mass and the angular velocities for each body, all relative to N, are given by

$$v = \begin{pmatrix} \mathbf{v}^{CMa} \\ {}^N\omega^A \\ \mathbf{v}^{CMb} \\ {}^N\omega^B \\ \mathbf{v}^{CMc} \\ {}^N\omega^C \end{pmatrix} = \begin{pmatrix} -{}^a t_3 \dot{q}_1 \mathbf{a}_2 + {}^a t_2 \dot{q}_1 \mathbf{a}_3 \\ \dot{q}_1 \mathbf{a}_1 \\ \left(-{}^b t_2 s_2 \dot{q}_1 + {}^b t_3 \dot{q}_2\right) \mathbf{b}_1 + \left({}^b t_1 s_2 \dot{q}_1 - {}^b t_3 c_2 \dot{q}_1\right) \mathbf{b}_2 + \left({}^b t_2 c_2 \dot{q}_1 - {}^b t_1 \dot{q}_2\right) \mathbf{b}_3 \\ \dot{q}_1 \mathbf{a}_1 + \dot{q}_2 \mathbf{b}_2 \\ {}^b l_1 s_2 \dot{q}_1 \mathbf{b}_2 - {}^b l_1 \dot{q}_2 \mathbf{b}_3 + \left({}^c t_2 \left(s_3 \dot{q}_2 - s_2 c_3 \dot{q}_1\right) + {}^c t_3 \left(s_2 s_3 \dot{q}_1 + c_3 \dot{q}_2\right)\right) \mathbf{c}_1 + \\ \left({}^c t_1 \left(s_2 c_3 \dot{q}_1 - s_3 \dot{q}_2\right) - {}^c t_3 \left(c_2 \dot{q}_1 + \dot{q}_3\right)\right) \mathbf{c}_2 + \left({}^c t_2 \left(c_2 \dot{q}_1 + \dot{q}_3\right) - {}^c t_1 \left(c_3 \dot{q}_2 + s_2 s_3 \dot{q}_1\right)\right) \mathbf{c}_3 \\ \dot{q}_1 \mathbf{a}_1 + \dot{q}_2 \mathbf{b}_2 + \dot{q}_3 \mathbf{c}_1 \end{pmatrix}$$

where $c_i = \cos(q_i)$ and $s_i = \sin(q_i)$. With the trivial kinematic differential equations $\dot{q}_i = u_i$ we find (by inspection) the matrix β, with tangent vectors as elements, as

$$\beta = \begin{pmatrix} -{}^a t_3 \mathbf{a}_2 + {}^a t_2 \mathbf{a}_3 & \mathbf{0} & \mathbf{0} \\ \mathbf{a}_1 & \mathbf{0} & \mathbf{0} \\ -{}^b t_2 s_2 \mathbf{b}_1 + \left({}^b t_1 s_2 - {}^b t_3 c_2\right) \mathbf{b}_2 + {}^b t_2 c_2 \mathbf{b}_3 & {}^b t_3 \mathbf{b}_1 - {}^b t_1 \mathbf{b}_3 & \mathbf{0} \\ \mathbf{a}_1 & \mathbf{b}_2 & \mathbf{0} \\ {}^b l_1 s_2 \mathbf{b}_2 + \left(-{}^c t_2 s_2 c_3 + {}^c t_3 s_2 s_3\right) \mathbf{c}_1 + & -{}^b l_1 \mathbf{b}_3 + \left({}^c t_2 s_3 + {}^c t_3 c_3\right) \mathbf{c}_1 & -{}^c t_3 \mathbf{c}_2 + {}^c t_2 \mathbf{c}_3 \\ \left({}^c t_1 \left(s_2 c_3\right) - {}^c t_3 c_2\right) \mathbf{c}_2 + \left({}^c t_2 c_2 - {}^c t_1 s_2 s_3\right) \mathbf{c}_3 & -{}^c t_1 s_3 \mathbf{c}_2 - {}^c t_1 c_3 \mathbf{c}_3 & \\ \mathbf{a}_1 & \mathbf{b}_2 & \mathbf{c}_1 \end{pmatrix}$$

For a detailed explanation of tangent vectors, see [7, 8].

2.2 Dynamics

To develop the dynamics, the velocity vectors (v) and tangent vectors (β) from the previous section are used. The momentum is the row matrix

$$p = v^T \bullet M = \begin{pmatrix} {}^a m \mathbf{v}^{CMa} & \mathbf{I}^{CMa} \bullet^N \boldsymbol{\omega}^A & {}^b m \mathbf{v}^{CMb} & \mathbf{I}^{CMb} \bullet^N \boldsymbol{\omega}^B & {}^c m \mathbf{v}^{CMc} & \mathbf{I}^{CMc} \bullet^N \boldsymbol{\omega}^C \end{pmatrix},$$

where M is a matrix with each rigid body along the diagonal and ${}^i m$ and $\mathbf{I}^{CMi}$ are the rigid body mass and moment of the inertia dyad, respectively. The first coloumn vector of p equals the linear momentum of body a, and the second coloumn vector of p equals the angular momentum of body a. The following elements of p describe the same momentum for body b and c. The applied forces on the system are generated by the gravitational acceleration and the motor torque in the joints

$$F_a = \begin{pmatrix} -{}^a m n_3 & \tau_1 \mathbf{a}_1 & -{}^b m n_3 & \tau_2 \mathbf{b}_2 & -{}^c m n_3 & \tau_3 \mathbf{c}_1 \end{pmatrix}$$

By differentiation of the momentum with respect to time relative to N, we obtain the equations of motions (projected onto the tangent vector space) as

$$\left(F_a - \frac{{}^N dp}{dt}\right) \bullet \beta = \begin{pmatrix} 0 & 0 & 0 \end{pmatrix}$$

Above, we have used d'Alembert principle of virtual work on the constraint forces which gives $F_c \bullet \beta = \begin{pmatrix} 0 & 0 & 0 \end{pmatrix}$. The projected equations of motions are written as

$$J_{ij}\ddot{q}_j + S_{ijk}\dot{q}_j\dot{q}_k + g_i = \tau_i \tag{1}$$

where we used free and dummy index notation and the summation convention, see Fung [4]. The individual terms in the equation above are given as follows:

$$\begin{aligned}
g_1 &= PG_1 s_1 s_2 + PG_2 s_1 c_2 s_3 + PG_3 s_1 c_2 c_3 + PG_4 s_1 c_2 + PG_5 s_1 + PG_6 c_1 + PG_7 c_1 s_3 + PG_8 c_1 c_3 \\
g_2 &= PG_9 s_2 s_3 c_1 + PG_{10} s_2 c_1 c_3 + PG_{11} s_2 c_1 + PG_{12} c_2 c_1 \\
g_3 &= PG_{13} s_1 c_3 + PG_{14} s_1 s_3 + PG_{15} c_1 c_2 s_3 + PG_{16} c_1 c_2 c_3 \\
J_{11} &= PJ_1 s_3^2 s_2^2 + PJ_2 s_3 c_3 s_2^2 + PJ_3 s_2^2 + PJ_4 s_2 c_2 s_3 + PJ_5 c_2 s_2 c_3 + PJ_6 c_2 s_2 + PJ_7 \\
J_{12} &= PJ_8 s_2 s_3^2 + PJ_9 s_2 s_3 c_3 + PJ_{10} s_2 + PJ_{11} c_2 s_3 + PJ_{12} c_2 c_3 + PJ_{13} c_2 \\
J_{13} &= PJ_{14} s_2 s_3 + PJ_{15} s_2 c_3 + PJ_{16} c_2 \\
J_{22} &= PJ_{17} c_3 s_3 + PJ_{18} c_3^2 + PJ_{19}, \quad J_{23} = PJ_{20} s_3 + PJ_{21} c_3, \quad J_{33} = PJ_{22}, \quad J_{31} = J_{13}, \quad J_{32} = J_{23} \\
S_{111} &= S_{121} = S_{131} = S_{132} = S_{212} = S_{221} = S_{222} = S_{223} = S_{231} = S_{232} = S_{313} = S_{321} = S_{323} = \\
&\quad S_{331} = S_{332} = S_{333} = 0 \\
S_{112} &= PS_1 s_3 c_3 s_2 c_2 + PS_2 c_3^2 s_2 c_2 + PS_3 s_2 c_2 + PS_4 c_2^2 s_3 + PS_5 c_2^2 c_3 + PS_6 c_2^2 + PS_7 s_3 + \\
&\quad PS_8 c_3 + PS_9 \\
S_{113} &= PS_{10} s_3 s_2 c_2 + PS_{11} c_3 s_2 c_2 + PS_{12} s_3 c_3 c_2^2 + PS_{13} c_2^2 c_3^2 + PS_{14} c_2^2 + PS_{15} s_3 c_3 + \\
&\quad PS_{16} c_3^2 + PS_{17} \\
S_{122} &= PS_{18} s_3 s_2 + PS_{19} c_3 s_2 + PS_{20} s_2 + PS_{21} s_3 c_3 + PS_{22} c_3^2 + PS_{23} c_2 \\
S_{123} &= PS_{24} s_3 c_3 s_2 + PS_{25} c_3^2 s_2 + PS_{26} s_2, \quad S_{133} = PS_{27} s_2 s_3 + PS_{28} s_2 c_3
\end{aligned}$$

$$
\begin{aligned}
S_{211} &= PS_{29}s_3c_3s_2c_2 + PS_{30}c_3^2s_2c_2 + PS_{31}s_2c_2 + PS_{32}c_2^2s_3 + PS_{33}c_2^2c_3 + PS_{34}c_2^2 + \\
&\quad PS_{35}s_3 + PS_{36}c_3 + PS_{37} \\
S_{213} &= PS_{38}s_3c_3s_2 + PS_{39}c_3^2s_2 + PS_{40}s_2, \;\; S_{233} = PS_{41}s_3 + PS_{42}c_3 \\
S_{311} &= PS_{43}s_3s_2c_2 + PS_{44}c_3s_2c_2 + PS_{45}c_3s_3 + PS_{46}c_3^2c_2^2 + PS_{47}c_2^2 + PS_{48}s_3c_3 + PS_{49}c_3^2 + PS_{50} \\
S_{312} &= PS_{51}s_3c_3s_2 + PS_{52}c_3^2s_2 + PS_{53}s_2 + PS_{54}c_2s_3 + PS_{55}c_2c_3 \\
S_{322} &= PS_{56}s_3c_3 + PS_{57}c_3^2 + PS_{58}
\end{aligned}
$$

The 96 parameters (PG, PJ and PS) above are defined by a set of physical parameters introduced in the model (see kinematics and dynamics above). The following 18 independent equations were found:

$$
\begin{aligned}
PG_1 &= ({}^cm\,{}^bl_1 + {}^c\,m\,{}^ct_1 + {}^b\,m\,{}^bt_1) \\
PG_2 &= -{}^cm\,{}^ct_2, \;\; PG_3 = -{}^cm\,{}^ct_3, \;\; PG_4 = -{}^bm\,{}^bt_3, \;\; PG_5 = -{}^am\,{}^at_3 \\
PG_6 &= {}^am\,{}^at_2 + {}^b\,m\,{}^bt_2 \\
PJ_1 &= {}^cJ_{22} - {}^c\,J_{33} + {}^c\,m(-{}^ct_2^2 + {}^c\,t_3^2), \;\; PJ_2 = 2({}^cJ_{23} - {}^c\,m\,{}^ct_2{}^ct_3) \\
PJ_3 &= {}^bJ_{33} - {}^c\,J_{11} - {}^b\,J_{11} + {}^c\,J_{33} + {}^b\,m({}^bt_1^2 - {}^b\,t_3^2) + {}^c\,m(2{}^bl_1{}^ct_1 + {}^b\,l_1^2 + {}^c\,t_1^2 - {}^c\,t_3^2) \\
PJ_4 &= 2({}^cJ_{12} - {}^c\,m\,{}^ct_2({}^bl_1 + {}^c\,t_1)), \;\; PJ_5 = 2({}^cJ_{13} - {}^c\,m\,{}^ct_2({}^bl_1 + {}^c\,t_1)), \;\; PJ_6 = 2({}^bJ_{13} - {}^b\,m\,{}^bt_1{}^bt_3) \\
PJ_7 &= {}^aJ_{11} + {}^b\,J_{11} + {}^c\,J_{11} + {}^a\,m({}^at_2^2 + {}^a\,t_3^2) + {}^b\,m({}^bt_2^2 + {}^b\,t_3^2) + {}^c\,m({}^ct_2^2 + {}^c\,t_3^2) \\
PJ_{10} &= {}^bJ_{23} - {}^c\,m\,{}^ct_2{}^ct_3 - {}^b\,m\,{}^bt_2{}^bt_3 + {}^c\,J_{23}, \;\; PJ_{13} = {}^b\,J_{12} - {}^b\,m\,{}^bt_1{}^bt_2, \;\; PJ_{16} = {}^c\,J_{11} + {}^c\,m({}^ct_2^2 + {}^c\,t_3^2) \\
PJ_{19} &= {}^bJ_{22} + {}^c\,J_{33} + {}^b\,m({}^bt_1^2 + {}^b\,t_3^2) + {}^c\,m({}^bl_1^2 + {}^c\,t_1^2 + {}^c\,t_2^2 + 2{}^bl_1{}^ct_1) \\
PS_3 &= 2\left(-{}^bJ_{11} + {}^b\,J_{33} - {}^c\,J_{11} + {}^c\,J_{22} + {}^b\,m({}^bt_1^2 - {}^b\,t_3^2) + {}^c\,m({}^bl_1^2 + 2{}^ct_1{}^bl_1 + {}^c\,t_1^2 - {}^c\,t_2^2)\right)
\end{aligned}
$$

3 Verification of Physical Parameters

Several authors have developed methods for identifying mechanical parameters, see for example [2, 5, 9, 11]. These papers, however, identify parameters of the type defined by the equations above, eg. PG_1 to PS_3, or the methods require force/torque sensors to be located at the robot's joints. In the following sections we describe the identification procedure in a practical setting at an industrial robot manufacturer. For most industrial robots, no external force/torque measurements are available. For the results presented here, only the motor torques and positions are measured. The joint velocities and acceleration data are obtained afterwards by filtering. Special care is taken to avoid phase shifts in velocity and acceleration, by using both forwards and backwards filtering of the position data.

As seen in section 2, the wrist model contains only 18 independent parameter equations, while 25 physical model parameters appear in the same equations. Hence, all these 25 physical parameters can not be solved simultaneously. However, the 18 independent equations can be used to verify linear and nonlinear combinations of the wrist's physical parameters obtained from CAD data. For example, if the identified parameter PG_2 and the corresponding estimated value from CAD data show large differences, the mechanical engineers get important feedback that either the value of ${}^cm$ or ${}^ct_2$ (or both) is wrong. The CAD data can then be modified or individual parts in a disassembled wrist can be measured more precisely using external measuring devices in an attempt to make the correspondance better. Having accurate CAD data is important for many reasons, including robot simulation studies and optimisation of motors and mechanical arm structures.

Note that the original wrist model illustrated in Figure 1 contains as many as 32 physical parameters. When modelling the wrist, 7 of these parameters (${}^at_1$, ${}^al_1$, ${}^aJ_{22}$, ${}^aJ_{33}$, ${}^aJ_{12}$, ${}^aJ_{13}$ and ${}^aJ_{23}$) disappear from the dynamic equations of motion. Hence, these 7 parameters are impossible to identify with the three motors configured as illustrated in Figure 1. When considering the complete 6-dof robot dynamic model, these 7 parameters appear in the equations and can potentially be identified. The problem of "invisible" parameters does, however, not disappear when considering the complete dynamics. As for the 3-dof dynamics, physical parameters belonging to the first axis of the robot will also disappear for the 6-dof dynamics.

The robot's control system uses the dynamic equations of motion to improve the wrist's path tracking accuracy. The 7 physical parameters which disappear from the equations will have no effect on the motion characteristics when only wrist motions is considered.

4 Identification of Model Parameters

In the previous sections we showed that it is possible to verify identified parameters against mechanical CAD data to discover errors in the modelled equations of motion. In this section, we identify 18 physical parameters by assuming that 7 of the physical wrist parameters are known in advance, for example by measurement before the wrist is assembled. The selection of these 7 parameters is not random. The physical parameters can be divided into three groups: 1) Masses and link lenghs, which are easy to measure, 2) mass centres and 3) inertia terms. When choosing the physical parameters which need to be measured before assembly, as many as possible should be from category 1) and 2) and as few as possible from category 3). In order to identify inertia terms before assembling the wrist, dedicated rotating machinery is required.

Given ${}^{a}m$,${}^{b}m$, ${}^{b}l_1$, ${}^{b}t_2$, ${}^{c}t_1$, the following solutions were found:

$$
\begin{aligned}
{}^{c}m &= \frac{-PG_2^2 + PG_3^2}{PJ_3 - \frac{1}{2}PS_3 + PJ_1} \\
{}^{c}t_2 &= -\frac{PG_2}{{}^{c}m},\ {}^{c}t_3 = -\frac{PG_3}{{}^{c}m},\ {}^{b}t_3 = -\frac{PG_4}{{}^{b}m},\ {}^{a}t_3 = -\frac{PG_5}{{}^{a}m} \\
{}^{a}t_2 &= \frac{1}{{}^{a}m}(PG_6 - {}^{b}m\,{}^{b}t_2),\ {}^{b}t_1 = \frac{1}{{}^{b}m}(PG_1 - {}^{c}m\,{}^{b}l_1 - {}^{c}m\,{}^{c}t_1) \\
{}^{c}J_{11} &= PJ_{16} - {}^{c}m({}^{c}t_2^2 + {}^{c}t_3^2),\ {}^{c}J_{23} = \frac{1}{2}PJ_2 + {}^{c}m\,{}^{c}t_2\,{}^{c}t_3 \\
{}^{b}J_{23} &= PJ_{10} + {}^{c}m\,{}^{c}t_2\,{}^{c}t_3 + {}^{b}m\,{}^{b}t_2\,{}^{b}t_3 - {}^{c}J_{23},\ {}^{b}J_{13} = \frac{1}{2}PJ_6 + {}^{b}m\,{}^{b}t_1\,{}^{b}t_3 \\
{}^{c}J_{12} &= \frac{1}{2}PJ_4 + {}^{c}m\,{}^{b}l_1\,{}^{c}t_2 + {}^{c}m\,{}^{c}t_1\,{}^{c}t_2,\ {}^{c}\mathrm{J}_{13} = \frac{1}{2}\mathrm{PJ}_5 + {}^{c}\mathrm{m}\,{}^{c}\mathrm{t}_1\,{}^{c}\mathrm{t}_3 + {}^{c}\mathrm{m}\,{}^{b}\mathrm{l}_1\,{}^{c}\mathrm{t}_3 \\
{}^{b}J_{12} &= PJ_{13} + {}^{b}m\,{}^{b}t_1\,{}^{b}t_2
\end{aligned}
$$

Given ${}^{b}J_{33}$ and ${}^{c}J_{33}$ we find the remaining 4 parameters:

$$
\begin{aligned}
{}^{c}J_{22} &= PJ_1 + {}^{c}J_{33} - {}^{c}m(-{}^{c}t_2^2 + {}^{c}t_3^2) \\
{}^{b}J_{22} &= PJ_{19} - {}^{c}J_{33} - {}^{b}m({}^{b}t_1^2 + {}^{b}t_3^2) - {}^{c}m({}^{b}l_1^2 + {}^{c}t_1^2 + {}^{c}t_2^2 + 2\,{}^{b}l_1^{c}t_1) \\
{}^{b}J_{11} &= -PJ_3 + {}^{b}J_{33} - {}^{c}J_{11} + {}^{c}J_{33} + {}^{b}m({}^{b}t_1^2 - {}^{b}t_3^2) + {}^{c}m(2\,{}^{b}l_1^{c}t_1 + {}^{b}l_1^2 + {}^{c}t_1^2 - {}^{c}t_3^2) \\
{}^{a}J_{11} &= -PJ_7 + {}^{b}J_{11} + {}^{c}J_{11} + {}^{a}m({}^{a}t_2^2 + {}^{a}t_3^2) + {}^{b}m({}^{b}t_2^2 + {}^{b}t_3^2) + {}^{c}m({}^{c}t_2^2 + {}^{c}t_3^2)
\end{aligned}
$$

The general solution given above, requires 2 masses, 1 link length, 2 mass centres and 2 inertia terms to be measured before assembling the wrist. The remaining 18 physical parameters can then be identified by excitation of the wrist motors on the final assembled wrist.

5 Experiments

Figure 2 shows measurements taken from the wrist motors which are transformed to link angles. To transform from motor variables to link side variables the following linear transformations are used

$$
\tau_a = G^T\tau_m, \quad \mathrm{q_m} = \mathrm{Gq_a} \qquad\qquad G = \begin{pmatrix} G_{11} & 0 & 0 \\ G_{21} & G_{22} & 0 \\ G_{31} & G_{32} & G_{33} \end{pmatrix}
$$

where τ_m is a 3×1 motor torque vector, τ_a is a 3×1 torque vector transformed to the link side and q_m and q_a are the corresponding motor and link angles, respectively. The matrix G is the transmission matrix. For a detailed description of the physical design of such wrist transmissions, see Craig [3], Figure 8.8, page 271. In addition to the rigid-body parameters which we have modelled as a function of the link variables in this paper, the friction appears and needs to be modelled and identified as well. Friction identification, however, is outside the scope of this paper. A good reference on friction modelling is Armstrong-Hélouvry[1]. If we include a simple friction model on the motor side consisting of Coloumb friction only, the motion dynamics on the arm side in (1) must be modified as follows.

$$
J_{ij}\ddot{q}_j + S_{ijk}\dot{q}_j\dot{q}_k + g_i + G_{ji}C_{jk}sgn(\dot{q}_k) = \tau_i
$$

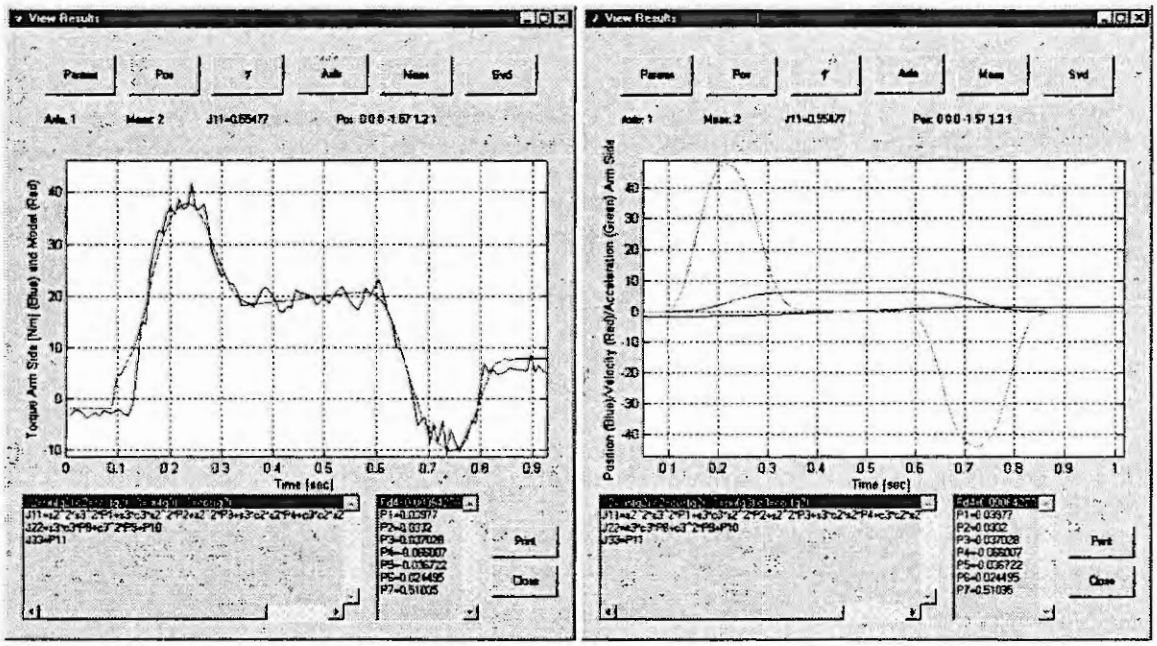


Figure 2: *Measurements from a 3-dof wrist of an ABB industrial robot. The figure on the left shows the measured torque for the first axis in Nm (noisy) and the corresponding modelled torque, given by equation (1). The figure on the right shows the measured position, velocity and acceleration for a ramp response in axis one.*

where C_{jk} is a diagonal 3×3 matrix with the Coloumb friction elements on the diagonal.

The identification of the physical parameters is divided into two steps. First, gravity, Coriolis, centripetal and inertia terms are identified for a given configuration of the wrist. Second, a series of identification results from step one are joined together to solve individual terms of the type PJ_1, PJ_2, etc. Some other solutions, for example [11], combine these two steps into one and perform the entire identification in one step. The advantage of dividing the identification into two steps, is that we can isolate individual physical terms, such as inertia, and check that we have persistent excitation for each of these terms.

The second step of the identification is a linear identification problem, and is well-known in the literature. Here we give a brief example of the identification of J_{11} in section 2.2. If we define the following vectors

$$P = [PJ_1\ PJ_2\ PJ_3\ PJ_4\ PJ_5\ PJ_6\ PJ_7]^T \text{ and } \phi = [s_3^2 s_2^2\ \ s_3 c_3 s_2^2\ \ s_2^2\ \ s_2 c_2 s_3\ \ c_2 s_2 c_3\ \ c_2 s_2\ \ 1] \quad (2)$$

then we can write $J_{11} = \phi P$. Let Φ be a matrix consisting of rows of different ϕ's from different excitation trajectories of the wrist. Then, the identification of P is simply given by $P = \Phi^+ J_{11}$, where $^+$ denotes the pseudo-inverse given by $A^+ = (A^T A)^{-1} A^T$, see [10, 11]. To get correct values for the parameter vector P, it is important to have persistent excitation in the trajectories. In other words, the matrix Φ needs a full rank, in this example a rank of 7. In our experiments, we defined 8 different wrist trajectories for the identification of J_{11} and computed the singular values of the matrix Φ. The singular values were found to be 3.63,1.28,0.86,0.73,0.59,0.31 and 0.30. From experimentation, we found the path trajectories to be acceptable when the smallest singular value of Φ was within approximately 10% of the largest singular value.

Figure 2 shows some of the final identified parameters. For example, the inertia of first axis of the wrist is identified as

$$J_{11} = 0.0398 s_3^2 s_2^2 + 0.0332 s_3 c_3 s_2^2 + 0.0370 s_2^2 - 0.0660 s_2 c_2 s_3 - 0.0367 c_2 s_2 c_3 + 0.0245 c_2 s_2 + 0.5104 \quad (3)$$

which gives 7 of the 25 parameters appearing in the 18 linearly independent equations, ie. $PJ_1 = 0.0398$, $PJ_2 = 0.0332$, $PJ_3 = 0.0370$, $PJ_4 = -0.0660$, $PJ_5 = -0.0367$, $PJ_6 = 0.0245$, $PJ_7 = 0.5104$. In addition

to these 7 parameters, the following parameters can also be identified by running experiments on the wrist: PG_1, PG_2, PG_3, PG_4, PG_5, PG_6, PJ_{10}, PJ_{13}, PJ_{16}, PJ_{19} and PS_3. Together with 7 physical parameters estimated before assembly of the wrist (${}^{a}m$,${}^{b}m$, ${}^{b}l_1$, ${}^{b}t_2$, ${}^{c}t_1$, ${}^{b}J_{33}$ and ${}^{c}J_{33}$), the remaining 18 physical parameters can all be identified.

6 Conclusions

In this paper we have demonstrated rigid-body dynamics modelling and methods for verification and identification of an industrial ABB robot wrist. The derivation of the equations was done in Maple using Sophia and Kane's method. With these tools, even the equations for complex 6-dof manipulators can be derived quickly.

For verification, linear combinations of the wrist parameters can be identified and compared with CAD data of the wrist. If the identified parameters and the CAD data show large differences, then the CAD data can be modified or individual parts in a disassembled wrist can be measured more precisely using external measuring devices in an attempt to make the correspondance better.

For identification, 32 physical parameters were included in the original model, where 25 of these appear in the dynamic equations of motion. Of these 25 parameters, we are able to identify 18 parameters, given initial estimates of the remaining 7 parameters. The methods presented in the paper have been tested with experiments on an ABB industrial wrist with internal transmission couplings. The identification methods are used at ABB today to improve a) the robots path tracking performance, b) the input data to dynamic simulation tools c) drive system optimization and the mechanical design of the arm structures.

References

[1] Armstrong-Hélouvry, B., *Control of Machines with Friction*, Kluwer Academic Publishers, 1991.

[2] Atkeson C.G., C.H. An and J.M. Hollerbach, "Estimation of Inertial Parameters of Manipulators Loads and Links", *International Journal of Robotics Research*, Vol. 5, No. 3, pp. 101-118, 1986.

[3] Craig, J.J., *Introduction to Robotics: Mechanics and Control*, J. J. Craig, Addison-Wesley, 1989.

[4] Fung, Y.C., *Foundations of Solid Mechanics*, Prentice-Hall, Inc., 1965.

[5] Gautier M., "Dynamic Identification of Robots with Power Model", *Proceedings of the 1997 IEEE International Conference on Robotics and Automation (ICRA '97)*, Albuquerque, 20-25 April 1997, pp. 1922-1927.

[6] Kane T.R. and D.A. Levinson, "*Dynamics: Theory and Applications*", McGraw-Hill 1985.

[7] Lennartsson A. "*Efficient Multibody Dynamics*", Doctorial Thesis, Royal Institute of Technology, Stockholm, Sweden, 1999. TRITA-MEK, Technical Report 1999:01, ISSN 0348-467X.

[8] Lesser M., "*The Analysis of Complex Nonlinear Mechanical System*", World Scientific Series on Nonlinear Science, Series A Vol.17, World Scientific Publishing Co. 1995.

[9] Reyes F. and R. Kelly, "On Parameter Identification of Robot Manipulators", *Proceedings of the 1997 IEEE International Conference on Robotics and Automation (ICRA '97)*, Albuquerque, 20-25 April 1997, pp. 1910-1915.

[10] Strang, G., *Linear Algebra and Its Applications*, Harcourt Brace Jovanovich, 1988.

[11] Swevers J., et.al., "Optimal Robot Excitation and Identification", *IEEE Transactions on Robotics and Automation*, Vol. 13, No. 5, Oct. 1997, pp. 730-740.

THE DEVELOPING OF MECHANICAL MODEL PARAMETERS FOR TIME DEPENDENT MATERIALS

K. Gotlih
University of Maribor, Faculty of Mechanical Engineering
Smetanova 17, SI-2000 Maribor, Slovenia
Gotlih@uni-mb.si

Abstract. The aim of this work is the determination and developing of mechanical model parameters for time dependent materials. Time dependent materials are those which do not have a unique response when the acting deformation or load changes its duration, amplitude or frequency. In most real cases the materials are the so-called visco-elastic materials. In this work we deal with textile materials and their basic carrying element, the thread. For modelling and simulation purposes of the textile structure the knowledge of the mechanical properties of the thread is very important. The developing of the mechanical model parameters for the thread is a combined experimental and numerical procedure. In the experimental part the real thread is analysed. Afterwards the results are used in an optimisation process for model parameter determination. In the last stage, the model's response is compared with the response of the real thread. The difference between these two responses is the measure for the accuracy of the developed model.

Introduction

Time dependent materials are materials with mechanical properties that respond differently to the change of the duration, amplitude or frequency of loads or deformations. In the field of mechanical engineering, materials that are most often used do not articulate the time dependency of mechanical properties. For this materials, in most cases metals, simple Hook's rheological material model is suitable [1]. Modern engineering materials and materials used in the textile industry are fibre structures like textiles or for more complex purpose composites. To use these materials as carrying structures the mechanical behaviour must be well-known. In the first stage the mechanical model of the real material must be chosen. In the theory of rheology [1-2] there are many different mechanical models. They can be divided into models for solids and models for fluids. Both can be further linear or non-linear. The purpose of models is to describe and simulate the real mechanical properties of the real material as well as possible and to be the connecting element between the real material and the mathematical abstract model. Mathematical models are in most cases ordinary or partial differential equations with parameters which must be known before in the simulations the model can be used.
In this work we try to obtain the mechanical properties of threads which are the basic carrying elements in the textiles. Thread is a special heterogen anisotropic one dimensional structure developed from fibres in a spinning process. Thread is also undertaken to special chemical treatments to achieve the required properties.
For the illustration let us introduce a special case for thread usage. In the vehicle production an interesting requirement for a seam on the seat cover appears. On seats with built-in side air-bags the side air-bag is mounted under the seat cover. In the case of an accident the side air-bag has to blow up to protect the passenger. To guarantee the protection the seam of the seat cover in front of the air-bag must tear. But in normal use the same seam should hold the parts of the seat cover together until the end of the vehicles life. For this reason it is very important to define exactly the strength of the used thread for the mentioned seam. To predict the strength of the thread and the seam the mechanical properties of thread must be known. In the first step the rheological model, that best covers the mechanical properties, must be chosen and then the parameters of the chosen model, which represents the mechanical properties of the material, are developed through an experimental and mathematical procedure.
In this work the Kelvin-Voigt rheological material model was chosen (1), Fig. 1. The parameters of this visco-elastic solid material model are the K and D. K is the elasticity constant and D is the damping constant of the material. ε is the specific deformation (the strain) and $\dot{\varepsilon}$ is the specific deformation rate (the strain rate). $F(t)$ is the resistant tension force which appears in the thread when applying tension on it.

Stress-strain properties with respect to different strain rates

Because of the visco-elasticity of the thread the tear experiment must be done with different strain rates. With respect to the ability of the measuring equipment in our laboratory the strain rates (relative deformation of the

specimen in percent per second) are chosen 0,2 %/s, 13,0%/s and 26,0%/s. The stress-strain behaviour for the mentioned strain rates is shown in Fig. 3.

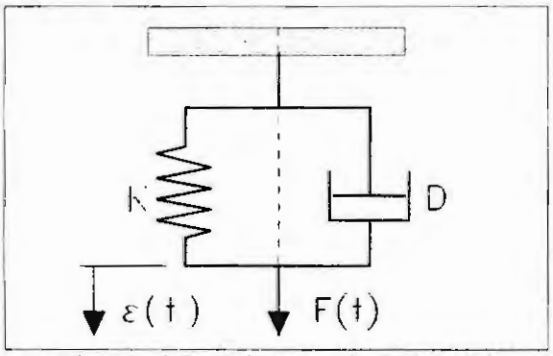


Fig. 1 Kelvin-Voigt's rheological model

Development of model parameters

The procedure is shown in the flow chart Fig. 2. After the experimental analysis of the real thread the data obtained are used in an optimisation procedure to compute the model parameters. The data used in the optimisation procedure are obtained from the stress-strain diagram given with respect to prescribed strain rate. The measured tension force in the real thread is the response of the thread on the acting deformation Fig. 3. The used strain rate is shown in Fig. 4. A standard routine (a non-linear optimisation procedure) from the numerical library IMSL, [3], which is connected with the Power station FORTRAN for PC computers under Windows 98, was used. The optimisation procedure uses the quasi Newton method. The gradients, which are used in the sensitivity analysis are computed with numerical finite difference method automatically. The cost functional (2) is chosen as a quadratic area under the curve which represents the difference between the real thread deformation and the response of the thread model (1) when the force *F(t)* is given with respect to time.

$$F(t) = D.\dot{\varepsilon} + K.\varepsilon \qquad (1)$$

$$I = \int_0^{\tau} (e_T - e_M)^2 dt \qquad (2)$$

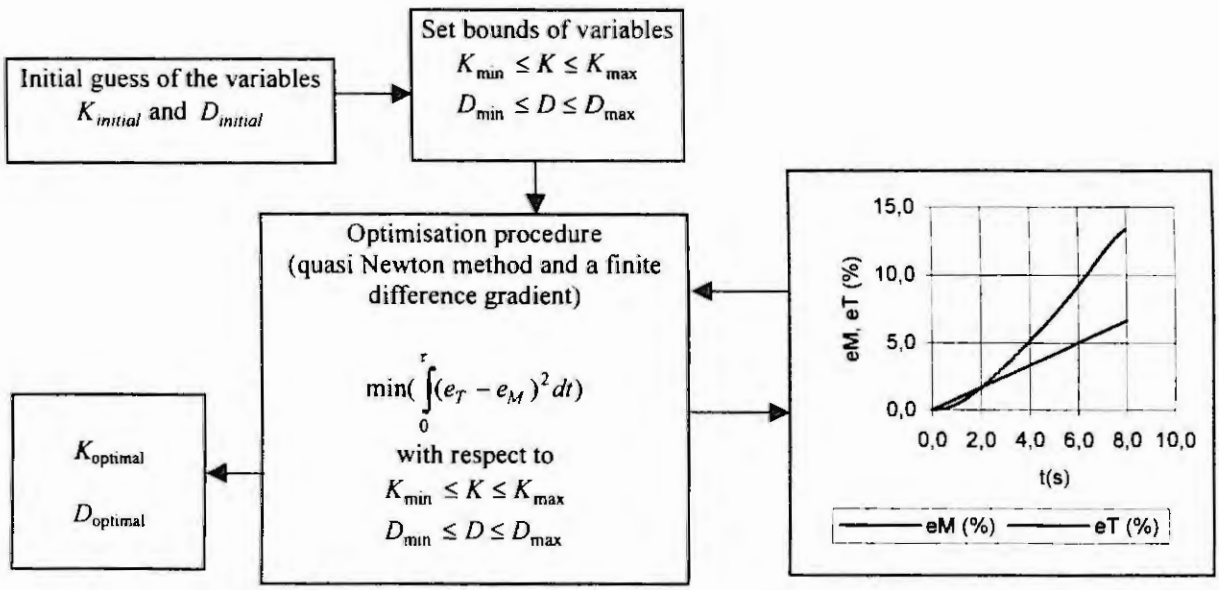


Fig. 2 Flow chart of the minimisation procedure
$e_M = \varepsilon_M$ (model) and $e_T = \varepsilon_T$ (real thread)

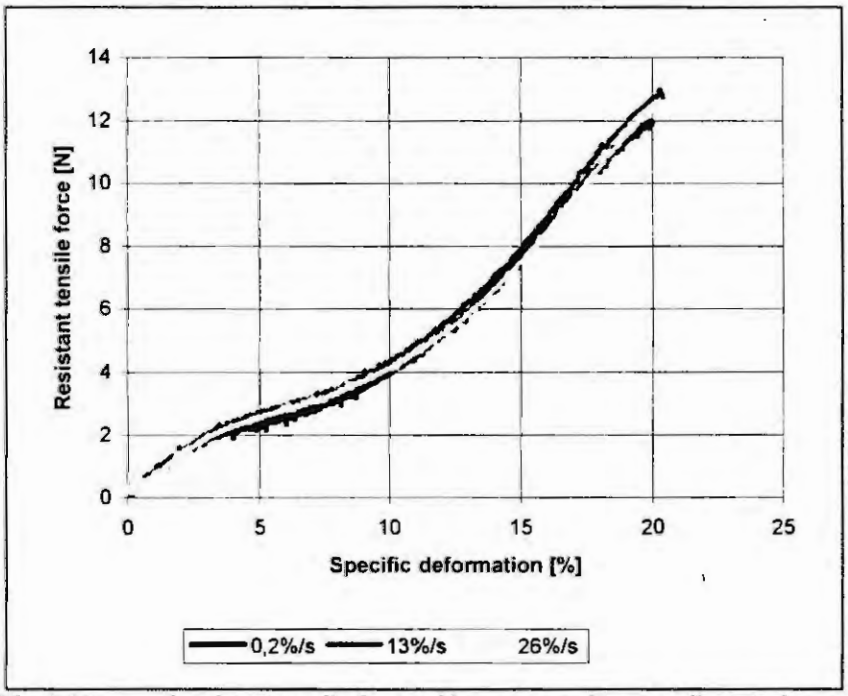


Fig. 3 Measured resistant tensile force with respect to the prescribed strain rate

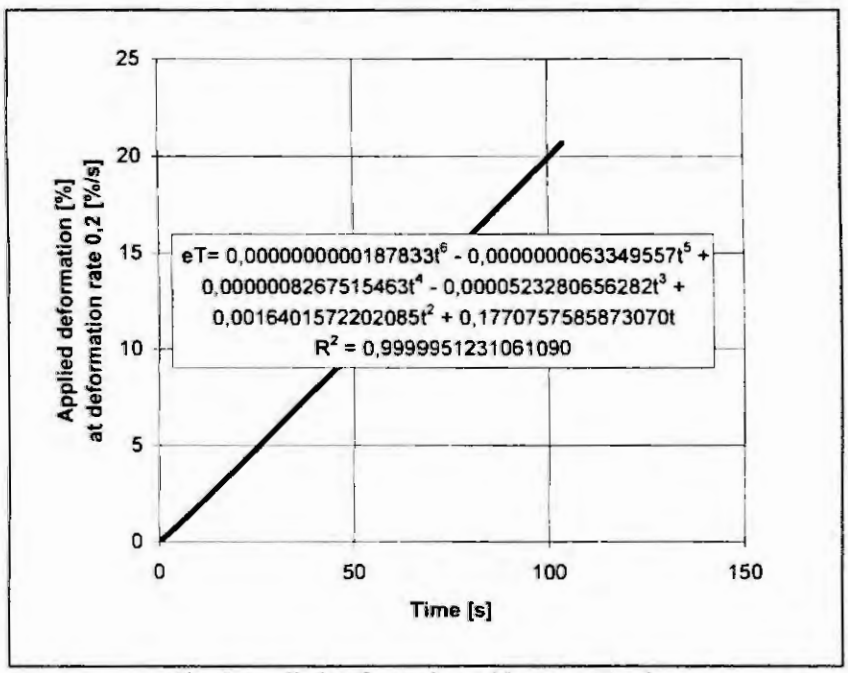


Fig. 4 Applied deformation with respect to time

Practical application

To illustrate the shown procedure a real thread with *28,97 tex* was chosen and the Kelvin-Voigt model parameters K and D were calculated. The computed constants are $D=21,0$ *Ns/%* and $K=0,01$ *N/%*. Differences between the real thread response and the model are shown in Fig. 6 for the strain rate *0,2 %/s*.

To show the suitability of the developed mathematical model (1), it was treated as a dynamic system. From Fig. 3 the response of the real thread on the applied deformation (Fig. 4) is known. A direct solving technique for the dynamic systems was used Fig. 5. If we know the applied deformation conditions on the real thread, the resistance tension force of the real thread, and the differential equation of the mathematical model (1), the

diagram in Fig. 6 can be drown. To solve the inital-value problem of the ordinary differential equation, Fig. 5, the Runge-Kutta-Verner fifth-order and sixth-order method, [3] was used.

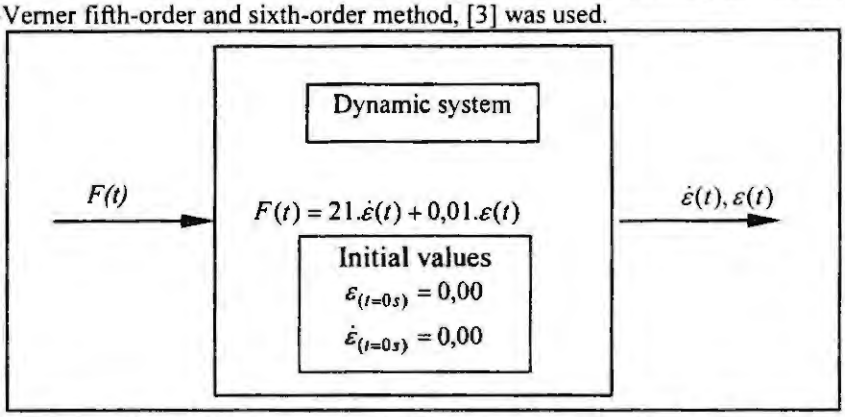


Fig. 5 Direct dynamical model for accuracy verification

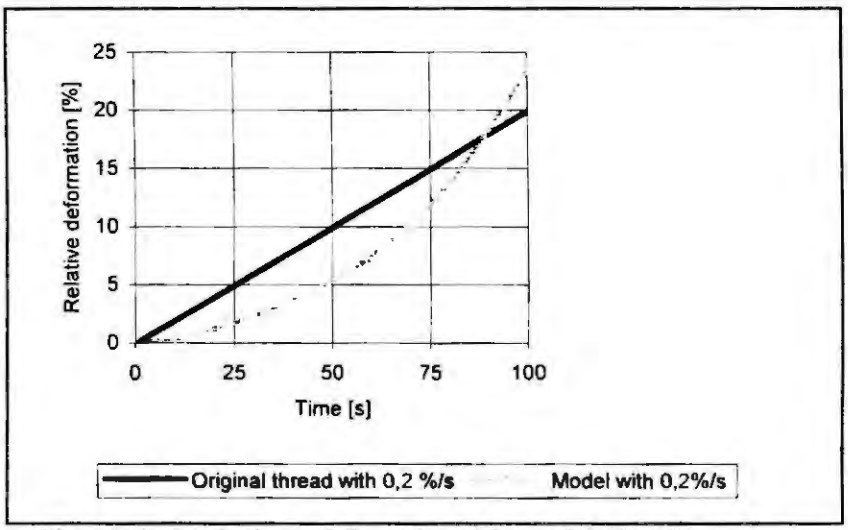


Fig. 6 Calculated relative deformation of the model, (1), with respect to the real resistant tensile forces $F(t)$, for deformation rates $\dot{\varepsilon}(t) = 0{,}2$ %/s

Conclusions

If we take in mind that thread is the carrying element of the seam, it becomes important to get mathematical model of it with the aim of better prediction and understanding of the stress-strain behaviour and the response of thread in the seam on applied loads or deformations. To get well improved mathematical models, the development must increase from simple, easy-to-understand and linear models to complicated, non-linear models. In this work the connection of experimental work with numerical methods is shown. The basis is the experimentally obtained deformation rate dependent stress-stain behaviour, that is connected with numerical methods (polynomial approximation, numerical optimisation and numerical solving of the initial value problem) to get a simple mathematical model in the form of ordinary differential equation.

References

1. Morton, W. E., Hearle, J. W. S., Physical Properties of Textile Fibres: The Textile Institute, Manchester, 1993.

2. Findley, W. N., Lai, J. S., Onaran, K., Creep and Relaxation of Non-linear Viscoelastic materials: North-Holland Publishing Company, Amsterdam, 1976.

3. IMSL/Library, Microsoft Fortran PowerStation Professional Edition, Ver 4.0, 1995.

OBJECT-ORIENTED HYBRID MODELLING OF MECHANICAL SYSTEMS

E. Carpanzano and L. Ferrarini
Dipartimento di Elettronica e Informazione, Politecnico di Milano
P.zza L. da Vinci 32, 20133 Milano, Italy
email: {carpanza,ferrarin}@elet.polimi.it

Abstract. An object-oriented framework for hybrid control systems modelling is presented. The proposed approach is based on a basic hybrid module, that can be recursively aggregated. Particular attention is paid to the formal definition of such a module, classifying its discrete state transitions in apparent, instantaneous and actual transitions. As an illustrative example, the modelling of some relevant mechanical hybrid phenomena is discussed.

1. Introduction

Mechanical systems show a dynamic behaviour that is typically affected by discontinuities and abrupt changes. Among the most common ones, there are the impact of a body on a rigid surface, friction phenomena in a rotational joint, or the passage from free motion to sliding motion and viceversa [4,8,9]. Other industrial areas encompass a large number of sub-systems with similar behaviour [1,7]. In order to properly model such hybrid phenomena, i.e. determined by the interaction of continuous and discrete-event dynamics, in the last years different techniques have been proposed. Moreover, in the vast majority of industrial plants, there are examples of discontinuities induced by the interaction between a physical process with its control system. To deal with this kind of phenomena a hybrid control system (HCS) model is needed [2,3,10].

In recent years, many modelling techniques have been proposed and used for different applications to deal with HCS [1,2,5,6,7]. In particular, it has been recognised in the technical literature that modularity with standardisation of interfaces, aggregation of simpler models with abstraction, model-ware reuse through inheritance, and strict correspondence of model objects to real objects, are the main features by which object-oriented modelling is superior in dealing with the complexity of real-size HCS [6,10,12]. In the present work, object-orientation is recognised as a key element for efficient modelling of a real-world HCS, and formal models are considered a crucial point to properly describe the hybrid phenomena dealt with [3]. Specifically, an object-oriented modelling technique is presented in sections 2 and 3. Furthermore, it is shown that important hybrid phenomena characterising the behaviour of mechanical systems can be clarified and formally described by means of the introduced framework (section 4). Finally, some concluding remarks concerning the benefits of the proposed technique and future work are drawn.

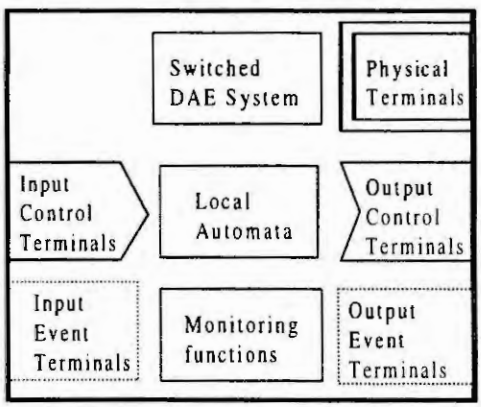


Fig. 1. Hybrid module for HCS modelling.

2. The hybrid module

The proposed modelling framework is based on the hybrid module shown in Fig. 1 [3,5]. The following three types of terminals constitute the module interfaces:

- physical terminals: (acausal physical connections through which power exchange takes place);
- control terminals: (causal connections through which information exchange takes place);
- event terminals, (causal propagation of events from one component to another component).

The formal description of the internal components of the module is given in the sequel.

Switched DAE system. This submodule holds a set of conditional DAEs with the following form, for a module **k**:

$\mathbf{F_k}(t, y_k, \dot{y}_k) = 0 \; ; \; \mathbf{S_{k1}}(t, y_k, \dot{y}_k) = 0 \; ; \; \mathbf{S_{k2}}(t, y_k, \dot{y}_k) = 0 \; ; \; \ldots \; ; \; \mathbf{S_{kn}}(t, y_k, \dot{y}_k) = 0$

where $\mathbf{S_{ki}}(t, y_k, \dot{y}_k) = 0$, $1 \le i \le n$, is defined by:

$f_{ki1}(t,y_k,\dot{y}_k) = 0$ while $\{M_{ki}=1\}$; or $f_{ki2}(t,y_k,\dot{y}_k) = 0$ while $\{M_{ki}=2\}$; ... or $f_{kiw_{ki}}(t,y_k,\dot{y}_k) = 0$ while $\{M_{ki}=w_{ki}\}$

In this system, $y_k(t)$ is a generally continuous real-valued vector function, $\mathbf{F_k}(t, y_k, \dot{y}_k) = 0$ is the subset of *fixed equations* of the module that always hold, $f_{kij}(t, y_k, \dot{y}_k) = 0$ is a *(conditional) vector equation* of the module that holds as long as $\{M_{ki} = j\}$, and M_{ki} is an integer variable whose value represents the current state of the i-th local automaton LA_{ki}.

Local automata. For each $\mathbf{S_{ki}}(t, y_k, \dot{y}_k) = 0$, a discrete-event model is defined: the local automaton $\mathbf{LA_{ki}}$, whose discrete states are numbered from 1 to w_{ki} and whose current state determines the vector equation that has to be considered among the set $f_{kij}(t, y_k, \dot{y}_k) = 0$, $1 \le j \le w_{ki}$.

More precisely, a local automaton LA_{ki} is a discrete-event model characterised by:

- a set of *states*, each of which is associated to (let the generic state be "j"): a *(conditional) vector equation* $f_{kij}(t, y_k, \dot{y}_k) = 0$, describing the dynamic behaviour of the subsystem corresponding to state j; and a *self-loop* with an associated *invariance condition*, mutually exclusive with all the transition conditions from that state, expressed by means of inequalities of the form $v(t, y_k, \dot{y}_k) \ge 0$ and Boolean operators;
- *directed edges* connecting different states, each one associated to: 1) a *mapping vector function* $\mathbf{g}^m_{kij}(t, y_k(t^-), \dot{y}_k(t^-), y_k(t^+), \dot{y}_k(t^+)) = 0$ to reinitialise the variables of the module when state m is reached from state j, if jumps of the variables are enforced by the event responsible for the state transition; 2) a *transition condition* and 3) a *backstep condition*, both expressed by means of events, inequalities of the form $v(t, y_k, \dot{y}_k) \ge 0$ and Boolean operators; condition 2) represents the conditions that enable the state transition between the current state and the connected one, while 3) is to be used with the new values of the variables $y_k(t^+)$ computed with the mapping function g^m_{kij}, in order to evaluate the admissibility of the new state.

For a local automaton LA_{ki} and for $\forall\, t \ge t_0$ an *equilibrium state* is defined as a state whose invariance condition is true at time t, with the assumption that there is always one (and only one) equilibrium state $\forall\, t \ge t_0$.

The behaviour of the local automata can be described as follows. Consider an automaton LA_{ki}, whose current equilibrium state is *e*, and suppose that the invariance condition of *e* becomes false. Then, all the directed edges stemming from the state *e* whose transition condition is true are considered. If the backstep condition for one of those directed edges holds, then the state transition associated to the directed edge does not occur. In this case, a "backstep" is said to happen. On the contrary, a new equilibrium state *n* is reached when the transition condition leading to it is true, and when the backstep condition results false. The first is expressed in terms of the variables at a time instant immediately before the event enabling the transition condition occurred (*a priori* values), and the second is expressed in terms of the variables computed through the mapping function g^n_e (*a posteriori* values). The complete and formal algorithm defining the automata evolution will be illustrated in detail in the next subsection, when dealing with the global model automaton. This will be done because the composition of hybrid modules can be described as a hybrid module as well, as shown in the following.

Monitoring functions: conditions involving events, inequalities of the form $h(t, y_k, \dot{y}_k) \ge 0$ and Boolean operators, which generate an event whenever any of their values changes.

In the described framework an *event* is generated whenever a transition condition, an invariance condition or a monitoring function change value, and is mathematically represented by means of a variable that is nonzero at its occurrence time instants.

3. The global model

Once all the modules describing the different subsystems of a HCS have been defined, the terminals of the different modules can be properly connected. The global model for the whole system can be described in turn through a hybrid module, without terminals, characterised by: the variables vector, given by the union of the variables of all the modules: $y = \cup_k y_k$; a set of fixed equations: the union of the sets of fixed equations of all the component modules: $\mathbf{F} = \cup_k \mathbf{F_k}$; the local automata $(\mathbf{LA_{ki}})$ of all modules; the binding equations $\mathbf{F_b}$ deriving

from the terminals connections; and a set of monitoring functions obtained by collecting the monitoring functions of all the component modules [3,5].

Now, since the local automata are all finite, the set of all the $\mathbf{LA_{ki}}$ is equivalent to a unique global model automaton, whose generic state α is defined by a combination of the states of all the local automata $\mathbf{LA_{ki}}$: $\alpha = <M_{11}, M_{12}, \ldots, M_{ki}, \ldots, M_{nm}>$. For each discrete state α of the global model automaton the set of *conditional equations* associated to α, and an *invariance condition* η_α can be defined. Moreover, for each directed edge from α to β the following elements can be introduced: the *mapping vector functions* $g_\alpha^\beta(\ldots)=0$ to reinitialise the variables of the model when β is reached from α; the *transition condition* $\gamma_\alpha^\beta(y(t))$ and the *backstep condition* $\mu_\alpha^\beta(y^+(t))$ from state α to state β.

Once the mentioned elements have been introduced, the *set of equations* $\mathbf{f}_\alpha$ describing the behaviour of the whole system in state α is given by: the fixed equations $\mathbf{F}$, the binding equations $\mathbf{F_b}$, and the set of conditional equations associated to α. In particular, the evolution of the global model can be described by the algorithm reported in the box, written in pseudo-Pascal (comments after the symbol ||).

```
Model evolution algorithm
beginalgo
  initialisation
    begin                          || begin initialisation
    find first state α0;
    solve the consistent initialisation problem for α0;
    α := α0;
    FOUND := true;
    end;                           || end initialisation
  repeat                           || begin main cycle
    while {ηα(y)=true} do:  || continuous evolution cycle
    {system evolves according to the set of equations fα};
    FOUND := false;
    while {(there are edges from α to be considered) AND
          (NOT FOUND)} do:         || new state searching
    { if {γαβ = true} then         || transition condition
      { compute y+ through gαβ;    || variable reinitialization
        if {μαβ(y+) = false} then  || backstep condition
        {       y = y+;            || variables updated
                α := β;            || discrete state updated
                FOUND:=true;}      || evolution in new state
      }                            || may start
    };
    until {NOT FOUND};             || end main cycle
    error;                         || no equilibrium found
endalgo.
```

According to the illustrated algorithm three distinct modes of operation can be distinguished: *apparent transition*, when the transition and the backstep conditions are true, in this case the new state β is said to be explored but not reached; *instantaneous transition*, when the transition condition is true, the backstep condition is false, and the invariance condition of the new state is false; in this case β is reached, the variables y are updated, but no continuous time evolution according to the new states equation occurs; whenever instantaneous transitions occur, more discrete states are "investigated" at the same point in time; and *actual transition*, when tbe transition condition from α to β is true, the backstep condition is false and the invariance condition for β is true; in this case the new state β is reached, the variables y are updated, and the system evolves according to the new state equations as long as the invariance condition of β holds. Phenomena involving a change of the systems equations are mathematically represented by actual state transitions, while phenomena involving variables jumps are mathematically described by instantaneous or actual state transitions. Apparent transitions do not represent a phenomenon characterising real HCS, since such transitions take place only in the model, and are typically due to imperfect knowledge of physical behaviour.

4. Illustrative example: modelling of hybrid mechanical phenomena

In [3] robotic system models are studied according to the presented modelling framework, since many examples can be found where hybrid problems are present and critical [4,8,9]. In particular, a robot arm is considered, which is realised through a six degree-of-freedom tree-structured multibody system, whose bodies are connected through rotational joints. The hybrid models for both the robot arm and its controller are discussed in detail in [3]. In Fig, 2 the modules and their connections for the whole HCS model are shown. In the following only the hybrid module for the generic link is discussed.

Consider a generic link composed of a rotational joint and of a rigid body. For the hybrid module representing the link, the following terminals can be defined: physical terminals to which horizontal and vertical

forces in the extremities, and the torques due to the joints motors are associated; an input control terminal to which the external torque τ_i is associated; an output control terminal to which the angular position of the body θ_i is associated.

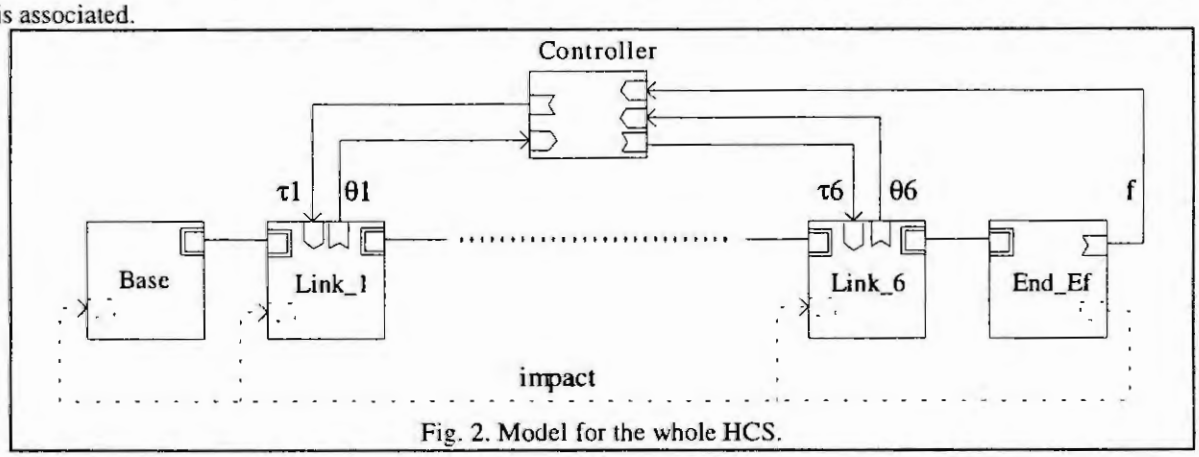


Fig. 2. Model for the whole HCS.

The classical dynamics equations describe the motion in the free space (Σ_{free}). If impacts have to be considered for the link, then the equations of the impulsive mechanics are necessary (Σ_{imp}). Now, the hybrid module describing the free motion and the impact dynamical behaviour of the i-th link can be defined through the local automaton LA_{dyn} depicted in Fig. 3. Moreover, for the considered module the following equations and functions can be introduced.

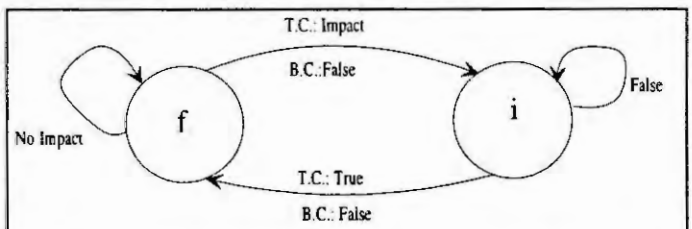


Fig. 3. Local automaton LA_{dyn} for the link dynamics.

Conditional equations for LA_{dyn}: Σ_{free} **while** {M = f }; **or** empty set **while** {M = i }

Mapping functions for LA_{dyn}: $g_f^i : \Sigma_{imp}$; $g_f^i : y_f = y_i$

As it is shown, the impulsive dynamics equations constitute the mapping function g_f^i of LA_{dyn}. Notice also that the transition from f to i in LA_{dyn} is instantaneous (since the invariance condition of i is false), which is in accordance with the impulsive mechanics.

Suppose now that it is required to introduce the joint friction in the link model. According to the proposed approach, the friction model can be easily introduced in the above defined hybrid module. Clearly, this enforces the modularity at the modelling level.

To describe the friction torque τ_f acting on the joint within the link module, the friction model represented in Fig. 4 is used [9]. In the model τ_f is the friction torque, τ_s is the stiction torque, τ_c is the Coulomb friction and $v_m = tg(\varphi)$ is the viscous friction coefficient. If the applied torque acting on the joint is given by τ_a then the model of Fig. 4 establishes that the friction torque can be computed through the following equations: Σ_1: $\tau_f = \tau_a$ while the link is motionless and $|\tau_a| \leq |\tau_s|$; Σ_2: $\tau_f = \tau_c\ sign(\dot{\theta}) + v_m\ \dot{\theta}$ while the link is moving; Σ_3: $\tau_f = \tau_c\ sign(\tau_a)$ when the link is motionless and τ_a becomes greater than τ_s. Such a model can be represented with the hybrid module for the link by simply adding the friction torque τ_f in the free motion dynamic equations, and by adding a new local automaton for the friction model. Such a local automaton, called LA_{fric}, and illustrated in Fig. 5, allows to compute the friction torque according to the model of Fig. 4. For this automaton the conditional equations and the mapping functions are the following ones.

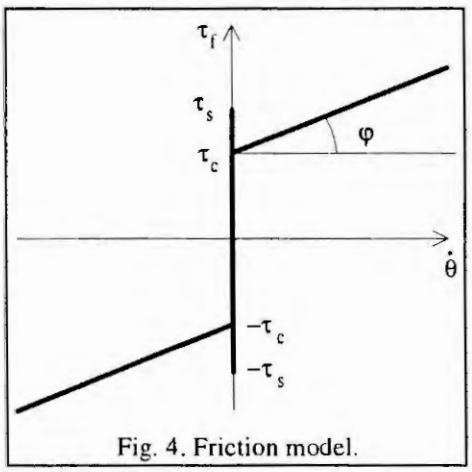


Fig. 4. Friction model.

Conditional equations: Σ_1 **while** $\{M_{fric} = \beta_1\}$; **or** Σ_2 **while** $\{M_{fric} = \beta_2\}$; **or** Σ_3 **while** $\{M_{fric} = \beta_3\}$

Mapping functions: $g_{\beta_1}^{\beta_3}$: $\tau_f = \tau_c\ sign(\tau_a)$; $g_{\beta_2}^{\beta_1}$: $\tau_f = \tau_a$; $g_{\beta_3}^{\beta_2}$: $\tau_f = \tau_c\ sign(\dot{\theta} + v_m\ \dot{\theta}$; $g_{\beta_2}^{\beta_3}$: $\tau_f = \tau_c\ sign(\tau_a)$

Notice that in LA$_{fric}$ an apparent transition is represented, which occurs when $\dot{\theta}$ becomes zero while state β_2 is active, and the corresponding backstep condition results true.

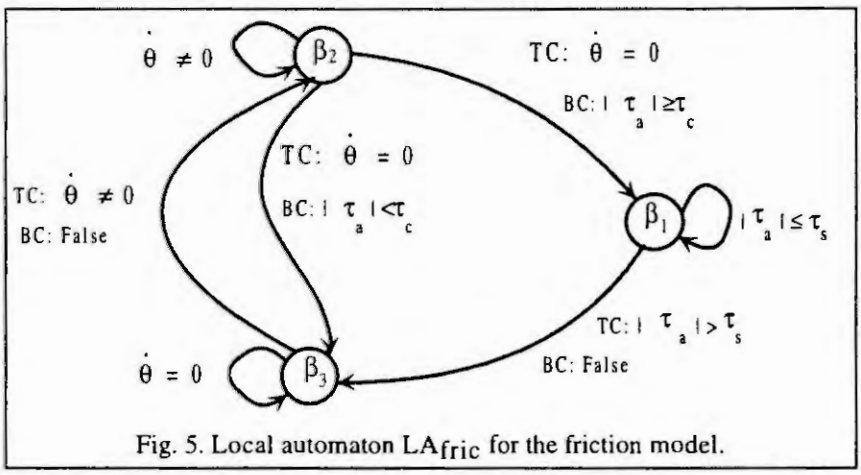


Fig. 5. Local automaton LA$_{fric}$ for the friction model.

5. Conclusions

In the paper a new object-oriented modelling technique for HCS representation is presented. The proposed framework is not conceived only for specific application domains, or for specific analysis or simulation purposes, like other known approaches [1,6,7]. Moreover, in [3,5] it is highlighted that all the different phenomena affecting HCS classified by Branicky et al. in [2] (autonomous switching, autonomous jumps, controlled switching, and controlled jumps) can be accurately described. In particular, a new classification of hybrid transition is given and formalised in apparent, instantaneous and actual transitions. In addition, the proposed technique allows to deal with the complexity of real-world HCS by exploiting the main features of the object-oriented approach. Actually, many of the existing object oriented modelling languages, e.g. Dymola [6], Omola, Modelica [12], MDL [11], provide syntax and semantics to represent HCS, but the formulation of automata and the propagation and management of events throughout the model have not yet been given a unified and coherent framework, though there are some proposals in literature [10]. In the model introduced here these aspects are studied in detail. Moreover, the proposed framework supports powerful abstraction mechanisms, since it is possible to (hierarchically) aggregate modules by properly connecting more modules, and model-ware reuse is strongly enforced, since it is easy to define new modules by specialising or refining previously defined models.

The actual work on the subject includes the application of the modelling approach to other complex industrial fields and the extension of the hybrid module definition in order to include more features of the control systems. On the other hand, future work will be the study of analysis and synthesis techniques for the proposed models, and the realisation of tools implementing such methods for control system design purposes.

References

1. Barton, P.I. and Pantelides, C.C., Modeling of Combined Discrete/Continuous Processes. AiChE J., Vol. 40, N° 6 (1994), 966-979.
2. Branicky, M., Borkar, V. and Mitter, S., A unified framework for hybrid control. In: Proc. IEEE Conf. on Decision and Control, Buena Vista, Florida, 1994, 4228-4234.
3. Carpanzano, E., A Development Methodology for Hybrid Control Systems. PhD Thesis, Politecnico di Milano, 1998.
4. Carpanzano, E., Fabbri, R. and Ferrarini, L., A Structured Methodology for the Design and Implementation of Hybrid Robot Controllers. In: Proc. IEEE Conf. on Control Applications, Trieste, Italy, 1998, 572-577.
5. Carpanzano, E. and Ferrarini, L., Modular Modelling of Hybrid Phenomena. In: Proc. European Control Conference, Karlsruhe, Germany, 1999, F654.
6. Elmqvist, H., Cellier, F.E. and Otter, M., Object Oriented Modelling of Hybrid Systems. In: Proc. European Simulation Symposium, Delft, 1993, 31-41.
7. Engell, S., Modelling and Analysis of Hybrid Systems. In: Proc. 2nd MathMod, Vienna, 1997, 17-31.
8. Ferrarini, L., Ferretti, G., Maffezzoni, C. and Magnani, G., Hybrid Modelling and Simulation for the Design of an Advanced Industrial Robots Controller. IEEE Robotics & Automation Magazine (1997), 45-51.
9. Ferretti, G., Maffezzoni, C. and Magnani, G, Dynamic Simulation of Robots Interacting with Stiff Contact Surfaces. Transactions of The Society for Computer Simulation, Vol. 9, N° 1 (1992), 1-24.
10. Maffezzoni, C., Ferrarini, L. and Carpanzano, E., Object-Oriented Models for Advanced Automation Engineering. Control Engineering Practice, 7 (1999), 957-968.
11. Maffezzoni, C. and Girelli, R., MOSES: Modular Modeling of Physical Systems in an Object-Oriented Database. Mathematical Modelling of Systems, Vol. 4, N° 2 (1998), 121-147.
12. Mattsson, S.E., Elmqvist, H. and Otter, M., Physical system modelling with Modelica. Control Engineering Practice, 6 (1998), 501-510.

KINEMATICS AND DYNAMICS OF CO-OPERATING MANIPULATORS ON A MOBILE BASE

Anjan Kumar Swain[1], Alan S. Morris[1] and Ali M.S. Zalzala[2]
[1] University of Sheffield, Sheffield S1 3JD, UK.
[2] Heriot-Watt University, Edinburgh, UK.

Abstract. This paper describes the kinematics and dynamics of multiple robotic arms mounted on a mobile base and co-operatively handling a common object. The presence of closed kinematic chains along with a mobile base of comparable mass and inertia with rest of the system makes the whole system dynamic analysis extremely complex. An attempt has been made to derive a complete unified dynamic model and the simulation algorithms for a general type of co-operative robotic system, with maximum emphasis on space robotic systems. All the derivations have been made in relation to space free-flying and free-floating robot manipulators.

1 Introduction and definition of notation

Co-operative robotic systems are an increasingly popular area of robotic systems research as they can perform tasks that cannot be easily handled by single manipulator. These tasks include handling large, heavy or non-rigid objects, assembly together of two or more separate components and space robotic applications. Analysis and control of co-ordinated systems is very complex due to the presence of inherent kinematic and dynamic interactions during execution of co-operative strategies.

In the case of fixed base co-operating manipulators, much work has been carried out [2, 4, 6]. However, many potential tasks in manufacturing and space robotics require the base to be mobile, and this makes the system even more complex. A mobile base with complete motion freedom, where the attitude can rotate about three axes as well as translate along spatial x, y and z axes, can be modelled as a six degree of freedom system in either free-flying or free-floating form. In the case of free-flying space robots, both the spacecraft and manipulator system are controlled simultaneously, whereas, once the spacecraft is positioned correctly, the spacecraft thruster system can be shut off to save fuel, and this is then known as a free-floating space robot.

The control of mobile-base robots is particularly challenging because of dynamic coupling when the mass and inertia of the base is comparable with that of the rest of the robotic system comprising the manipulators and the object handled. Past research on mobile-base systems has usually only involved a single manipulator, but some work has been reported recently on multiple manipulator systems [3], although this only covers the free-floating case and not the free-flying one.

This paper describes the derivation of a unified generalised dynamic model for a co-operating robotic system mounted on a mobile base of comparable mass and inertia, where the latter is subject to an external force. Unlike previous work, the model covers both the free-floating and free-flying cases. Simpler models can be readily derived from this generalised formulation. For example, for a fixed-base co-operating system, the base has zero mobility and infinite mass and inertia, and insertion of the appropriate base parameters produces a simplified model.

The general model of a co-ordinated robot manipulator system with m-robots, each with n_i links, installed on a moving base, is shown in Fig.1. The end-effectors hold a common object rigidly and it is assumed that the total mass and inertia of the robotic manipulators and the object is comparable with that of the mass and inertia of the base. To make the system more generalised, the model includes an external spatial force $\mathbf{f}_b$ applied to the base to represent the spacecraft thruster force that is met in free-flying space-robotics applications.

The co-ordinate frames are defined according to a modified form of the Denavit-Hartenberg convention such that the co-ordinate frame of a particular link is attached to that link with frame origin at the near end of the link. The spatial velocity, acceleration and force vectors of the ith link of the jth robot resolved in the ith link frame are denoted by (6×1) vectors ${}^{i}\mathbf{V}_j$, ${}^{i}\dot{\mathbf{V}}_j$ and ${}^{i}\mathbf{f}_j$. The 6 × 6 spatial transformation matrix ${}^{i}_{i-1}\mathbf{X}_j$ transforms a spatial vector from (i-1)th co-ordinate frame to the ith co-ordinate frame of the jth robot and is defined as [1]:

$$ {}^{i}_{i-1}\mathbf{X}_j = \begin{bmatrix} {}^{i}_{i-1}\mathbf{R}_j & \mathbf{0} \\ {}^{i}_{i-1}\mathbf{R}_j\,{}^{i}\tilde{\mathbf{p}}_j^{T} & {}^{i}_{i-1}\mathbf{R}_j \end{bmatrix} \quad \text{where:} \quad \tilde{\mathbf{p}} = \begin{bmatrix} 0 & -p_z & p_y \\ p_z & 0 & -p_x \\ -p_y & p_x & 0 \end{bmatrix} $$

and ${}^{i}_{i-1}\mathbf{R}_j$ is a 3 × 3 rotation matrix from the (i-1)th link frame to the ith link frame for the jth robot; ${}^{i}_{i-1}\mathbf{p}_j$ is a 3 × 1 vector from the origin of the (i-1)th link frame to the origin of the ith link frame for the jth robot.

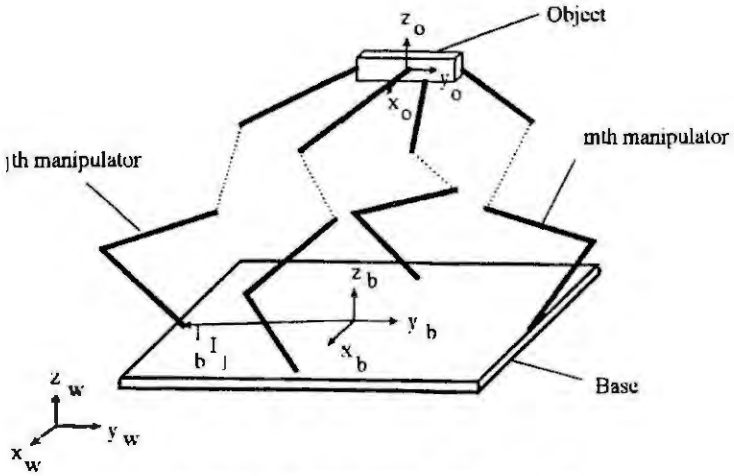


Fig 1: A co-operating robot system

The 6 × 6 spatial inertia matrix of the ith link of the jth robot is denoted by ${}^{i}\mathbf{M}_j$ and is defined as [1]:

$$ {}^{i}\mathbf{M}_j = \begin{bmatrix} -{}^{i}m_j\,{}^{c}_{i}\tilde{\mathbf{l}}_j & {}^{i}m_j\mathbf{E} \\ {}^{i}\mathbf{I}_j & {}^{i}m_j\,{}^{c}_{i}\tilde{\mathbf{l}}_j \end{bmatrix} $$

where ${}^{i}m_j$ is the mass of the ith link of the jth robot; ${}^{i}\mathbf{I}_j$ is the inertia tensor of the ith link of the jth robot at the ith frame origin; ${}^{c}_{i}\mathbf{l}_j$ is the distance from the ith frame origin to the centre of mass of the ith link of the jth robot; and $\mathbf{E}$ is an identity matrix.

2 Development of generalised model

Using orthogonal vectors ${}^{i}\Phi_j$ and ${}^{i}\Phi^{c}_j$ to represent the matrix of free and constrained mode vectors of the ith joint of the jth robot, the spatial velocity of the ith link of the jth robot represented in the ith link frame can be expressed in terms of the velocity of the base $\mathbf{V}_b$ as [5]:

$$ {}^{i}\mathbf{V}_j = {}^{i}_{b}\mathbf{X}_j\,\mathbf{V}_b + \sum_{k=1}^{i} {}^{i}_{k}\mathbf{X}_j\,{}^{k}\Phi_j\,{}^{k}\dot{\mathbf{q}}_j + \sum_{k=1}^{i} {}^{i}_{k}\mathbf{X}_j\,{}^{k}\xi_j = {}^{i}_{b}\mathbf{X}_j\,\mathbf{V}_b + \sum_{k=1}^{i} {}^{i}_{k}\mathbf{X}_j\,({}^{k}\Phi_j\,{}^{k}\dot{\mathbf{q}}_j + {}^{k}\xi_j) \qquad (1) $$

The spatial velocity of all links of all m robots can be expressed as: $\mathbf{V} = \mathbf{X}_b\,\mathbf{V}_b + \mathbf{X}(\Phi\,\dot{\mathbf{q}} + \xi)$ (2)

where $\mathbf{V} = [\mathbf{V}_1^T\ \mathbf{V}_2^T \ldots \mathbf{V}_m^T]^T$, $\mathbf{X}_b = [{}_b\mathbf{X}_1^T\ {}_b\mathbf{X}_2^T \ldots {}_b\mathbf{X}_m^T]^T$, $\mathbf{X} = \mathrm{diag}(\mathbf{X}_1\ \mathbf{X}_2 \ldots \mathbf{X}_m)$, $\Phi = \mathrm{diag}(\Phi_1\ \Phi_2 \ldots \Phi_m)$, $\dot{\mathbf{q}} = [\dot{\mathbf{q}}_1^T\ \dot{\mathbf{q}}_2^T \ldots \dot{\mathbf{q}}_m^T]^T$, and $\xi = [\xi_1^T\ \xi_2^T \ldots \xi_m^T]^T$.

The end-effector velocity of the jth robot, denoted by $\mathbf{v}^{e}_j$, can be expressed in the base frame as:

$$ \mathbf{v}^{e}_j = {}_b\mathbf{R}_j\,{}^{n+1}_{b}\mathbf{X}_j\,\mathbf{V}_b + {}_b\mathbf{R}_j\,{}^{n+1}_{n}\mathbf{X}_j\,\mathbf{L}_j(\Phi_j\,\dot{\mathbf{q}}_j + \xi_j) = \mathbf{J}_{bj}\,\mathbf{V}_b + \mathbf{J}_{qj}\,\dot{\mathbf{q}}_j + \mathbf{k}_j\,\xi_j \qquad (3) $$

where ${}_b\mathbf{R}_j = \mathrm{diag}({}^{b}_{n+1}\mathbf{R}_j\ {}^{b}_{n+1}\mathbf{R}_j)$ with ${}^{b}_{n+1}\mathbf{R}_j$ is a rotation matrix from the end-effector frame to the base frame, $\mathbf{k}_j = {}_b\mathbf{R}_j\,{}^{n+1}_{n}\mathbf{X}_j\,\mathbf{L}_j$, $\mathbf{J}_{qj} = {}_b\mathbf{R}_j\,{}^{n+1}_{n}\mathbf{X}_j\,\mathbf{L}_j\,\Phi_j$, and $\mathbf{J}_{bj} = {}_b\mathbf{R}_j\,{}^{n+1}_{b}\mathbf{X}_j$.

For all the m-robots, Eq.(3) can be represented as: $\mathbf{v}^{e} = \mathbf{J}_b\,\mathbf{V}_b + \mathbf{J}_q\,\dot{\mathbf{q}} + \mathbf{k}\,\xi$ (4)

The momentum of the entire system with respect to the inertial reference frame can be expressed as

$$ \mathbf{P} = ({}_w\mathbf{M}_b + {}_w\mathbf{M}_o\,\mathbf{J}'_b + \mathbf{M}_w\mathbf{T}\,\mathbf{X}_b)\,\mathbf{V}_b + (\mathbf{M}_w\mathbf{T}\,\mathbf{X}\,\Phi + {}_w\mathbf{M}_o\,\mathbf{J}'_q)\,\dot{\mathbf{q}} + (\mathbf{M}_w\mathbf{T}\,\mathbf{X} + {}_w\mathbf{M}_o\,\mathbf{k}')\,\xi \qquad (5) $$

For a free-floating space robotic system, the total momentum of the system P will be zero, therefore from Eq.(5) the base velocity can be expressed as:

$$\mathbf{V}_b = -({}_w\mathbf{M}_b + {}_w\mathbf{M}_o\,\mathbf{J}'_b + \mathbf{M}_w\mathbf{T}\,\mathbf{X}_b)^{-1}\{(\mathbf{M}_w\mathbf{T}\,\mathbf{X}\,\Phi + {}_w\mathbf{M}_o\,\mathbf{J}'_q)\dot{\mathbf{q}} + (\mathbf{M}_w\mathbf{T}\,\mathbf{X} + {}_w\mathbf{M}_o\,\mathbf{k}')\,\xi\}$$
$$= \mathbf{J}_r\,\dot{\mathbf{q}} + \mathbf{k}_1\,\xi \qquad (6)$$

where $\mathbf{J}_r = -({}_w\mathbf{M}_b + {}_w\mathbf{M}_o\,\mathbf{J}'_b + \mathbf{M}_w\mathbf{T}\,\mathbf{X}_b)^{-1}(\mathbf{M}_w\mathbf{T}\,\mathbf{X}\,\Phi + {}_w\mathbf{M}_o\,\mathbf{J}'_q)$ and

$\mathbf{k}_1 = -({}_w\mathbf{M}_b + {}_w\mathbf{M}_o\,\mathbf{J}'_b + \mathbf{M}_w\mathbf{T}\,\mathbf{X}_b)^{-1}(\mathbf{M}_w\mathbf{T}\,\mathbf{X} + {}_w\mathbf{M}_o\,\mathbf{k}')$.

The m end-effector velocity vector of a free-floating robotic system can be derived using Eqs.(4) and (6):

$$\mathbf{v}^e = (\mathbf{J}_b\,\mathbf{J}_r + \mathbf{J}_q)\dot{\mathbf{q}} + (\mathbf{J}_b\,\mathbf{k}_1 + \mathbf{k})\,\xi = \mathbf{J}\,\dot{\mathbf{q}} + \mathbf{k}_e\,\xi \qquad (7)$$

For free-flying robotic systems, the above relationship is not valid since momentum is not conserved, and the external force $\mathbf{f}_b$ becomes very important. This can be expressed as the rate of change of total momentum $\mathbf{P}$:

$$\mathbf{f}_b = \dot{\mathbf{P}} = ({}_w\mathbf{M}_b + {}_w\mathbf{M}_o\,\mathbf{J}'_b + \mathbf{M}_w\mathbf{T}\,\mathbf{X}_b)\dot{\mathbf{V}}_b + (\mathbf{M}_w\mathbf{T}\,\mathbf{X}\,\Phi + {}_w\mathbf{M}_o\,\mathbf{J}'_q)\ddot{\mathbf{q}} + \mathbf{b}_r \qquad (8)$$

where $\mathbf{b}_r$ is the net bias force acting on the system, which is a function of position, velocity and time.

The acceleration of the manipulators links can be obtained by differentiating Eq.(2) to give:

$$\dot{\mathbf{V}} = \mathbf{X}\,\Phi\,\ddot{\mathbf{q}} + \dot{\mathbf{X}}_b\,\mathbf{V}_b + \mathbf{X}_b\,\dot{\mathbf{V}}_b + \zeta \quad \text{where } \zeta = \dot{\mathbf{X}}(\Phi\,\dot{\mathbf{q}} + \xi) + \mathbf{X}(\dot{\Phi}\,\dot{\mathbf{q}} + \dot{\xi}).$$

The force exerted on the ith link of the jth robot is expressed as:

$${}^i\mathbf{f}_j = {}^i_{n+1}\mathbf{X}_j\,{}^{n+1}\mathbf{f}_j + \sum_{k=i}^{n} {}^i_k\mathbf{X}_j\,({}^k\mathbf{M}_j\,{}^k\dot{\mathbf{V}}_j + {}^k\mathbf{b}_j) \qquad (9)$$

Now the force vector for the jth robot can be described as: $\mathbf{f}_j = \mathbf{X}_j^T(\mathbf{D}_j\,{}^{n+1}\mathbf{f}_j + \mathbf{M}_j\,\dot{\mathbf{V}}_j + \mathbf{b}_j)$, where

$\mathbf{f}_j = [{}^1\mathbf{f}_j^T\,{}^2\mathbf{f}_j^T \ldots {}^n\mathbf{f}_j^T]^T$, $\dot{\mathbf{V}}_j = [{}^1\dot{\mathbf{V}}_j^T\,{}^2\dot{\mathbf{V}}_j^T \ldots {}^n\dot{\mathbf{V}}_j^T]^T$, $\mathbf{b}_j = [{}^1\mathbf{b}_j^T\,{}^2\mathbf{b}_j^T \ldots {}^n\mathbf{b}_j^T]^T$, $\mathbf{D}_j = [0 \ldots 0\ {}^{n+1}_n\mathbf{X}_j]^T$ and $\mathbf{M}_j$ $= \mathrm{diag}({}^1\mathbf{M}_j\,{}^2\mathbf{M}_j \ldots {}^n\mathbf{M}_j)$. For all the m robots this force relationship can be expressed as:

$$\mathbf{f} - \mathbf{X}^T\,\mathbf{D}\,\mathbf{f}_e = \mathbf{X}^T(\mathbf{M}_q\,\dot{\mathbf{V}} + \mathbf{b}) \qquad (10)$$

where $\mathbf{f} = [\mathbf{f}_1^T\,\mathbf{f}_2^T \ldots \mathbf{f}_m^T]^T$, $\dot{\mathbf{V}} = [\dot{\mathbf{V}}_1^T\,\dot{\mathbf{V}}_2^T \ldots \dot{\mathbf{V}}_m^T]^T$, $\mathbf{b} = [\mathbf{b}_1^T\,\mathbf{b}_2^T \ldots \mathbf{b}_m^T]^T$, $\mathbf{D} = \mathrm{diag}(\mathbf{D}_1\,\mathbf{D}_2 \ldots \mathbf{D}_m)$, $\mathbf{M}_q = \mathrm{diag}(\mathbf{M}_1\,\mathbf{M}_2 \ldots \mathbf{M}_m)$, and $\mathbf{f}_e$ is the force exerted by the end-effector on the object, which can be expressed by the relationship $\mathbf{f}_e = [{}^{n+1}\mathbf{f}_1^T\,{}^{n+1}\mathbf{f}_2^T \ldots {}^{n+1}\mathbf{f}_m^T]^T$.

From Eq.(9) the force exerted by the base on the first link of the jth robot can be expressed as:

$${}^1\mathbf{f}_j = {}^1_{n+1}\mathbf{X}_j\,{}^{n+1}\mathbf{f}_j + \sum_{k=1}^{n} {}^1_k\mathbf{X}_j\,({}^k\mathbf{M}_j\,{}^k\dot{\mathbf{V}}_j + {}^k\mathbf{b}_j) \qquad (11)$$

The force equilibrium equation of the base with an external force acting on it, can be represented as

$$\mathbf{f}_b = \mathbf{M}_b\,\dot{\mathbf{V}}_b + \mathbf{b}_b + \mathbf{X}_b^T(\mathbf{D}\,\mathbf{f}_e + \mathbf{M}_q\,\dot{\mathbf{V}} + \mathbf{b}) \qquad (12)$$

Now solving Eq.(12) for the base acceleration and substituting the value of $\dot{\mathbf{V}}_b$, the acceleration of the manipulators can be expressed as

$$\dot{\mathbf{V}} = (\mathbf{E} + \mathbf{X}_b\,\mathbf{M}_b^{-1}\,\mathbf{X}_b^T\,\mathbf{M}_q)^{-1}\{\mathbf{X}\,\Phi\,\ddot{\mathbf{q}} + \dot{\mathbf{X}}_b\,\mathbf{V}_b - \mathbf{X}_b\,\mathbf{M}_b^{-1}\,\mathbf{X}_b^T(\mathbf{D}\,\mathbf{f}_e + \mathbf{b}) + \mathbf{X}_b\,\mathbf{M}_b^{-1}(\mathbf{f}_b - \mathbf{b}_b) + \zeta\} \qquad (13)$$

Eliminating $\dot{\mathbf{V}}$ from Eqs.(10) and (13) gives:

$$\mathbf{f} - \mathbf{X}^T(\mathbf{M}_q^{-1} + \mathbf{X}_b\,\mathbf{M}_b^{-1}\,\mathbf{X}_b^T)^{-1}\mathbf{M}_q^{-1}\,\mathbf{D}\,\mathbf{f}_e = \mathbf{X}^T(\mathbf{M}_q^{-1} + \mathbf{X}_b\,\mathbf{M}_b^{-1}\,\mathbf{X}_b^T)^{-1}\{\mathbf{X}\,\Phi\,\ddot{\mathbf{q}} + \dot{\mathbf{X}}_b\,\mathbf{V}_b + \mathbf{M}_q^{-1}\,\mathbf{b} + \mathbf{X}_b\,\mathbf{M}_b^{-1}(\mathbf{f}_b - \mathbf{b}_b) + \zeta\} \qquad (14)$$

where $(\mathbf{M}_q^{-1} + \mathbf{X}_b\,\mathbf{M}_b^{-1}\,\mathbf{X}_b^T)^{-1} = \mathbf{M}_q(\mathbf{E} + \mathbf{X}_b\,\mathbf{M}_b^{-1}\,\mathbf{X}_b^T\,\mathbf{M}_q)^{-1}$.

Multiplying both the sides of the Eq.(14) with Φ^T and rearranging leads to a concise representation of the joint torque vector as:

$$\mathbf{T} - \mathbf{J}^T\,\mathbf{f}_e = \mathbf{M}\,\ddot{\mathbf{q}} + \mathbf{C} \qquad (15)$$

where $\mathbf{M} = \Phi^T\,\mathbf{X}^T(\mathbf{M}_q^{-1} + \mathbf{X}_b\,\mathbf{M}_b^{-1}\,\mathbf{X}_b^T)^{-1}\,\mathbf{X}\,\Phi$, $\mathbf{J}^T = \Phi^T\,\mathbf{X}^T(\mathbf{M}_q^{-1} + \mathbf{X}_b\,\mathbf{M}_b^{-1}\,\mathbf{X}_b^T)^{-1}\,\mathbf{M}_q^{-1}\,\mathbf{D}$,

$\mathbf{C} = \Phi^T\,\mathbf{X}^T(\mathbf{M}_q^{-1} + \mathbf{X}_b\,\mathbf{M}_b^{-1}\,\mathbf{X}_b^T)^{-1}\{\dot{\mathbf{X}}_b\,\mathbf{V}_b + \mathbf{M}_q^{-1}\,\mathbf{b} + \mathbf{X}_b\,\mathbf{M}_b^{-1}(\mathbf{f}_b - \mathbf{b}_b) + \zeta\}$, and $\mathbf{T} = \Phi^T\,\mathbf{f}$.

The inversion of the matrix in the Eq.(14) can be simplified using a matrix inversion lemma:

$$(\mathbf{M}_q^{-1}+\mathbf{X}_b\,\mathbf{M}_b^{-1}\,\mathbf{X}_b^T)^{-1} = \mathbf{M}_q - \mathbf{M}_q\,\mathbf{X}_b(\mathbf{M}_b + \mathbf{X}_b^T\,\mathbf{M}_b\,\mathbf{X}_b)^{-1}\,\mathbf{X}_b^T\,\mathbf{M}_q$$

If the object is assumed to be held rigidly by m manipulators, then the force at the centre of mass of the object due to all the end-effector forces acting on it, can be represented as [7]: $\mathbf{f}_o = \mathbf{W}^T\,\mathbf{f}_e$ (16)

The force balance equation for this object can be represented as: $\mathbf{f}_o = \mathbf{M}_o\,\dot{\mathbf{v}}^o + \mathbf{b}_o$ (17)

where $\mathbf{b}_o$ is the bias force on the object. Combining Eqs.(16) and (17) now leads to the following dynamic equation for the object: $\mathbf{M}_o\,\dot{\mathbf{v}}^o + \mathbf{b}_o = \mathbf{W}^T\,\mathbf{f}_e$ (18)

The forward dynamics analysis of the co-operating manipulators on a mobile base can be described with reference to Eq.(15), which involves the computation of the joint accelerations $\ddot{\mathbf{q}}$ with the knowledge of the input torques and forces, $\mathbf{T}$, current state of the manipulator, $\mathbf{q}$, $\dot{\mathbf{q}}$ and motion of the base. Simplifying Eq.(8) for $\dot{\mathbf{V}}_b$ and carrying out further simplification leads to the following expression for the end effector acceleration $\dot{\mathbf{v}}^e$:

$$\begin{aligned}\dot{\mathbf{v}}^e = {} & \{\mathbf{J}_q - \mathbf{J}_b\,\mathbf{M}_b^{-1}\,\mathbf{X}_b^T\,\mathbf{M}_q(\mathbf{X}\,\Phi - \mathbf{X}_b\,\mathbf{G}\mathbf{H})\}\,\ddot{\mathbf{q}} + (\dot{\mathbf{J}}_b - \mathbf{J}_b\,\mathbf{M}_b^{-1}\,\mathbf{X}_b^T\,\mathbf{M}_q\,\dot{\mathbf{X}}_b)\,\mathbf{V}_b \\ & + \mathbf{J}_b\,\mathbf{M}_b^{-1}(\mathbf{E} - \mathbf{X}_b^T\,\mathbf{M}_q\,\mathbf{G})\,\mathbf{f}_b + \mathbf{J}_b\,\mathbf{M}_b^{-1}(\mathbf{X}_b^T\,\mathbf{M}_q\,\mathbf{G}\mathbf{b}_r - \mathbf{b}_b - \mathbf{X}_b^T\,\mathbf{b} + \zeta) \\ & + (\dot{\mathbf{J}}_q\,\dot{\mathbf{q}} + \mathbf{k}\,\dot{\xi} + \dot{\mathbf{k}}\,\xi) - \mathbf{J}_b\,\mathbf{M}_b^{-1}\,\mathbf{X}_b^T\,\mathbf{D}\,\mathbf{f}_e \\ = {} & \dot{\mathbf{v}}^e_{open} - \dot{\mathbf{v}}^e_{constrained}\end{aligned} \quad (19)$$

Hence, the system can be modelled as a superposition of open chain part and a constrained part due to the presence of Cupertino. The following explicit relationship between the end-effector force and object acceleration can now be obtained:

$$\dot{\mathbf{v}}^e_{constrained} = \dot{\mathbf{v}}^e_{open} - \mathbf{W}\,\dot{\mathbf{v}}^o - \dot{\mathbf{W}}\,\mathbf{v}^o \quad \Rightarrow \mathbf{f}_e = (\mathbf{J}_b\,\mathbf{M}_b^{-1}\,\mathbf{X}_b^T\,\mathbf{D})^{-1}\,[\dot{\mathbf{v}}^e_{open} - \mathbf{W}\,\dot{\mathbf{v}}^o - \dot{\mathbf{W}}\,\mathbf{v}^o] \quad (20)$$

Then, substituting the value of $\mathbf{f}_e$ from Eq.(20) into Eq.(18) gives:

$$\dot{\mathbf{v}}^o = [\mathbf{M}_o + \mathbf{W}^T(\mathbf{J}_b\,\mathbf{M}_b^{-1}\,\mathbf{X}_b^T\,\mathbf{D})^{-1}\,\mathbf{W}]^{-1}\,\{\mathbf{W}^T(\mathbf{J}_b\,\mathbf{M}_b^{-1}\,\mathbf{X}_b^T\,\mathbf{D})^{-1}\;(\dot{\mathbf{v}}^e_{open} - \dot{\mathbf{W}}\,\mathbf{v}^o) - \mathbf{b}_o\} \quad (21)$$

Once the spatial acceleration of the object $\dot{\mathbf{v}}^o$ is known, Eq.(20) can give all the end-effector spatial forces. Similarly from Eq.(15), $\ddot{\mathbf{q}}$ can be represented as: $\ddot{\mathbf{q}} = \mathbf{M}^{-1}(\mathbf{T} - \mathbf{C}) - \mathbf{J}^T\,\mathbf{f}_e = \ddot{\mathbf{q}}_{open} - \ddot{\mathbf{q}}_{constrained}$ (22)

Hence, the joint accelerations of the entire system can be computed with the known open chain joint accelerations and end effector forces.

3 Conclusions

In this paper, a unified approach for the kinematics and dynamics of a co-operating robotic system on a mobile base has been presented, with special emphasis on space robotics. In the presence of closed kinematic chain constraints, the kinematics and dynamics of space robots become increasingly complex. Both inverse dynamics and forward dynamics of these systems have been addressed. In addition, it has been shown that the simulation of this type of system can be carried out using any efficient multi-arm unconstrained analysis approaches.

References

[1] Featherstone, R., *Robot Dynamics Algorithms*, 1987, Kluwer Academic Publishers, USA.

[2] Hu, Y-R., Goldenberg, A.A. and Zhou, C., "Motion and force control of co-ordinated robots during constrained motion task," *Int. J. Robotics Res*, 1995, vol. 14, no. 4, pp. 351-365.

[3] Hu, Y-R and Vukovich, G., "Dynamics of free-floating co-ordinated space robots," *J. Robotics Systems*, 1998, vol. 15, no. 4, pp. 217-230.

[4] Lilly, K.W. and Orin, D.E., "Efficient dynamic simulation of multiple chain robotic mechanisms," *J. Dynamic Syst., Measurement & Control*, 1994, vol. 116, pp. 223-231.

[5] Roberson, R.E. and Schwertassek, R, *Dynamics of Multibody Systems*, 1988, Springer-Verlag.

[6] Rodriguez, G., "Recursive forward dynamics for multiple robot arms moving a common task object," *IEEE Trans. Robotics & Autom*, 1989, vol. 5, no. 4, pp. 510-521.

[7] Wen, J.T. and Kreutz-Delgado, K., "Motion and force control of multiple robotic manipulators," *Automatica*, 1992, vol. 28, no. 4, pp. 729-743.

CONTINUOUS MODELLING OF ROBOT DYNAMICS USING A MULTI-DIMENSIONAL RBF-LIKE NEURAL NETWORK

M. Krabbes and C. Döschner
Otto-von-Guericke-University Magdeburg, Institute of Automation
PF 4120, D-39016 Magdeburg, Germany http://ifatwww.et.uni-magdeburg.de

Abstract. An identification approach of manipulator dynamics by means of a neural architecture is presented. In a structured model approach, a RBF-like neural network is used to represent and adapt all model parameters with their nonlinear dependences on the joint positions. The neural architecture is hierarchically organised to reach optimal adjustment to the common structural knowledge about the identification problem. A fixed, grid based neuron placement together with application of B-spline polynomial basis functions is utilised favourably for a very effective recursive implementation. That way an online identification of a dynamic model is submitted for a complete 6 joint industrial robot with reasonable effort and good results.

Introduction

It is well known, that best results in control of 6 joint robot arms are achievable by using an (inverse) model of robot dynamics due to the not negligible nonlinearities and couplings in the dynamic behaviour of the individual axes. However, preparation of such a model applicable in the control loop is still problematic, because on one hand even very complex analytic arrangements do not cover all occurring physical effects but on the other hand an exact estimation of all the particular parameters is very complicated. As an alternative to these methods, some approaches of modelling based on different supervised trained neural networks are discussed in recent years [1,3,5]. Based on these works, this contribution proposes a special neural Radial Basis Function (RBF)-like architecture for the modelling with good capabilities for both, a close estimation of the real phenomena and a continuous adaption of the performing model without difficulties. Available knowledge from theoretical process analysis is applied in a *structured* model, whose parameters are represented by the outputs of a neural network.

Robot Dynamics

Expression of the common inverse dynamic equations of a manipulator describes the relation between the torques in the robot joints $\vec{\tau}$ and the current movement state consisting of the joint positions $\vec{\theta}$ and its derivatives $\dot{\vec{\theta}}$.

$$\vec{\tau} = \mathbf{A}(\vec{\theta})\ddot{\vec{\theta}} + \mathbf{C}(\vec{\theta},\dot{\vec{\theta}}) + \mathbf{G}(\vec{\theta}) + \mathbf{F}(\vec{\theta},\dot{\vec{\theta}}) \tag{1}$$

$\mathbf{A}(\vec{\theta})$ is the inertia matrix, $\mathbf{C}(\vec{\theta},\dot{\vec{\theta}})$ the matrix of centrifugal and coriolis terms, and $\mathbf{G}(\vec{\theta})$ is the vector of gravity terms. $\mathbf{F}(\vec{\theta},\dot{\vec{\theta}})$ represents all friction effects, which are modelled by $\mathbf{F}(\vec{\theta}) \cdot \vec{f}_F(\dot{\vec{\theta}})$ with Coulomb, viscous, and declining parts [3]. In a comparable manner to the friction model, also the velocity-dependent part of the term $\mathbf{C}(\vec{\theta},\dot{\vec{\theta}})$ can be written split off in $\mathbf{C}(\vec{\theta}) \cdot \vec{f}_C(\dot{\vec{\theta}})$. Thus, the complete *state space equation* is converted to a *configuration space equation* since the matrices are (nonlinear !) functions of manipulator position only [2]:

$$\vec{\tau} = \mathbf{A}(\vec{\theta})\ddot{\vec{\theta}} + \mathbf{C}(\vec{\theta})\vec{f}_C(\dot{\vec{\theta}}) + \mathbf{G}(\vec{\theta}) + \mathbf{F}(\vec{\theta})\vec{f}_F(\dot{\vec{\theta}}) \quad \text{with: } \vec{f}_C(\dot{\vec{\theta}}) = [diag\{\dot{\vec{\theta}}\}^{\mathsf{T}} \cdot \dot{\vec{\theta}}]^{\mathsf{T}};\ \vec{f}_F(\dot{\vec{\theta}}) = [sign(\dot{\vec{\theta}}), \dot{\vec{\theta}}, \dot{\vec{\theta}}^{2/3}]^{\mathsf{T}} \tag{2}$$

I.e., the nonlinear dependences of $\mathbf{C}$ and $\mathbf{F}$ on the current state $[\vec{\theta},\dot{\vec{\theta}}]^{\mathsf{T}}$ are separated into the analytical couplings resp. functions of $\dot{\theta}$ and purely θ-dependent nonlinearities. They should be represented by a neural network, because this is able to take also further, unexpected effects into consideration by its error-minimising learning abilities. It is appropriate to consider the overall model error, because then Least-Square-based methods become applicable for the parameter estimation. Hence, the following summarised, parameter-linear function is proposed:

$$\vec{\tau}_{n\times 1} = \mathbf{ACGF}(\vec{\theta})_{n\times l} \cdot \vec{h}_{\theta\, l\times 1} \quad \text{with: } \vec{h}_\theta = [\ddot{\theta}_i, \dot{\theta}_i\dot{\theta}_j, 1, sign(\dot{\theta}_i), \dot{\theta}_i, \dot{\theta}_i^{2/3}]^{\mathsf{T}} \tag{3}$$

Model Reduction

A lot of reductions in the matrix $\mathbf{ACGF}(\vec{\theta})$ are accessible by general reconsideration. This simplifications effectuate, that particular parameters depend only on a subset of all joint positions up to constant or disappearing parameters. Moreover, there are some parameters, which can be calculated immediately from other ones:

As a physical fact the inertia matrix **A** is symmetric. Furthermore, the serial chaining of arm segments implicates, that any rotational inertia A_{ij}, which affects along axis i, is not depending on the position of axis i and all axes before it $k < i$. Thereby not only the symmetric **A**-matrix is strongly reduced, but also matrix **C** by analysis using the universally valid relation $C_{i,jk}(\vec{\theta}) = \frac{1}{2}(\partial A_{ij}(\vec{\theta})/\partial\theta_k + \partial A_{ik}(\vec{\theta})/\partial\theta_j - \partial A_{jk}(\vec{\theta})/\partial\theta_i)$ [4].

Additionally, the following specificities of the utilised SIEMENS manutec r2, representing common industrial robots, require further consideration: (i) The mass distribution of the last segment is considered as rotational symmetric. Hence, there is no dependence of inertias on the position of them. (ii) The vector $\mathbf{G}(\vec{\theta})$ is reduced, because the first axis is oriented in vertical direction and the last axis is a rotational symmetric one. (iii) In the 3 matrices $\mathbf{F}_*$, which are of diagonal type, only the gravity effects of the 2nd and 3rd arm segment are regarded to cause (also with respect to the load) notably position dependent friction parameters. (iv) Because of the orthogonal posture of the axes some elements of **A** are (exactly or approximately) disappearing ($A_{1,3}$, $A_{2,4}$, $A_{3,4}$, $A_{4,5}$, $A_{5,6}$). (v) As a typical construction feature of todays articulated robots all drives of the wrist joints are mounted on the 3rd arm segment. So they are mechanically coupled and appear as a closed kinematic chain. To keep the symmetry of **A** unaffected, only the rotor inertias of the motors 5 and 6 are regarded concerning the high gear transmission ratios. But in the matrices of **F**, parameters also appear outside of the diagonals $F_{i,j}$; $i \neq j$; $i, j \geq 4$.

i	$A_{i,1}$	$A_{i,2}$	$A_{i,3}$	$A_{i,4}$	$A_{i,5}$	$A_{i,6}$	$F_{*i,1}$	$F_{*i,2}$	$F_{*i,3}$	$F_{*i,4}$	$F_{*i,5}$	$F_{*i,6}$
1	$f(2,3,4,5)$	$f(2,3,4,5)$	0	$f(2,3,4,5)$	$f(2,3,4,5)$	$f(2,3,4,5)$	$f(2,3)$	0	0	0	0	0
2	sym.	$f(3,4,5)$	$f(3,4,5)$	0	$f(3,4,5)$	$f(3,4,5)$	0	$f(2,3)$	0	0	0	0
3	0	sym.	$f(4,5)$	0	$f(4,5)$	$f(4,5)$	0	0	$f(2,3)$	0	0	0
4	sym.	0	0	$f(5)$	0	0	0	0	0	const.	const.	const.
5	sym.	sym.	sym.	0	coost.	0	0	0	0	const.	const.	const.
6	sym.	sym.	sym.	0	0	const.	0	0	0	const.	const.	const.

$Ci,11$	$C_{i,12}$	$C_{i,13}$	$C_{i,14}$	$C_{i,15}$	$C_{i,16}$	$C_{i,22}$	$C_{i,23}$	$C_{i,24}$	$C_{i,25}$	$C_{i,26}$
0	$f(2,3,4,5)$	$f(2,3,4,5)$	$f(2,3,4,5)$	$f(2,3,4,5)$	0	$f(2,3,4,5)$	$f(2,3,4,5)$	$f(2,3,4,5)$	$f(2,3,4,5)$	$f(2,3,4,5)$
calc.	0	$f(2,3,4,5)$	$f(2,3,4,5)$	$f(2,3,4,5)$	$f(2,3,4,5)$	0	$f(3,4,5)$	$f(3,4,5)$	$f(3,4,5)$	0
calc.	calc.	0	$f(2,3,4,5)$	$f(2,3,4,5)$	$f(2,3,4,5)$	calc.	0	$f(3,4,5)$	$f(3,4,5)$	$f(3,4,5)$
calc.	calc.	calc.	0	$f(2,3,4,5)$	$f(2,3,4,5)$	calc.	calc.	0	$f(3,4,5)$	$f(3,4,5)$
calc.	calc.	calc.	calc.	0	$f(2,3,4,5)$	calc.	calc.	calc.	0	$f(3,4,5)$
0	calc.	calc.	calc.	calc.	0	0	0	0	0	0

$C_{i,33}$	$C_{i,34}$	$C_{i,35}$	$C_{i,36}$	$C_{i,44}$	$C_{i,45}$	$C_{i,46}$	$C_{i,55}$	$C_{i,56}$	$C_{i,66}$	G
$f(2,3,4,5)$	$f(2,3,4,5)$	$f(2,3,4,5)$	$f(2,3,4,5)$	$f(2,3,4,5)$	$f(2,3,4,5)$	$f(2,3,4,5)$	$f(2,3,4,5)$	$f(2,3,4,5)$	0	0
$f(3,4,5)$	$f(3,4,5)$	$f(3,4,5)$	$f(3,4,5)$	0	$f(3,4,5)$	$f(3,4,5)$	$f(3,4,5)$	$f(3,4,5)$	0	$f(2,3,4,5)$
0	$f(4,5)$	$f(4,5)$	0	0	$f(4,5)$	$f(4,5)$	$f(4,5)$	$f(4,5)$	0	$f(2,3,4,5)$
calc.	0	$f(4,5)$	$f(4,5)$	0	$f(5)$	0	0	0	0	$f(2,3,4,5)$
calc.	calc.	0	$f(4,5)$	calc.	0	calc.	0	0	0	$f(2,3,4,5)$
0	calc.	calc.	0	0	calc.	0	calc.	0	0	0

Table 1: Dependences of $\mathbf{ACGF}(\vec{\theta})$ ($f(i) \cong f(\theta_i)$, $\mathbf{F}_*$ appears tripled, the order is changed for better reading).

Table 1 shows, that there are no remaining dependences on θ_1 or θ_6 and there are only a few different ascending cases of $f(*)$. These characteristics are consequently used by representation of each coefficient merely within the according subspace of Θ under advantageous utilisation of a the networks grid-based architecture.

The Neural Network Architecture

The multi-dimensional nonlinear function $\mathbf{ACGF}(\vec{\theta})$ can be mapped suitably by the sensory-motor transformation of a RBF-like neural architecture [6]. Because the neural network is utilised within a control loop, a sufficiently exact representation of the whole input space has to be guaranteed. Therefore the neurons are attached to regular positions in a right-angular grid. The architecture proposed in this work takes important advantages from this orthogonal and invariant neuron placement, because the hierarchical structure is build up by means of recursive implementation. That way it is managed to represent every parameter of $\mathbf{ACGF}(\vec{\theta})$ within one network exactly in a Θ-subspace of its dependences. Thereby the fact is utilised, that all different cases of parameter dependence occur in an ascending series from $f(\theta_5)$ to $f(\theta_2, \theta_3, \theta_4, \theta_5)$ (beside $f(\theta_2, \theta_3)$, see Table 1).

Any neuron consists of its usual output weights $\vec{W}$ and of a second vector, which defines the input weights θ_i of the *accessory* dimension in the subsequent hierarchical layer (Fig. 1, l.). The hierarchical structure of the network is build up based on single neuron in the first layer, that represents all constant parameters. The second vector of this neuron defines the one-dimensional configuration of the chosen number of neurons for the first parameter dependence. So the neurons of the following layer at these positions represent the θ_5-dependent parameters and, again, arrange the neurons of the next layer with an added dimensionality and so on (Fig. 1, m.,r.). For such parameters, which have $f(\theta_2, \theta_3)$-dependence, an extra network is used with usual two-dimensional grid structure.

It is necessary that a continuous map is generated from the support nodes at the neuron positions. This is done in RBF-networks by a weighted superposition of radial-symmetrical basis functions, which are centred at the

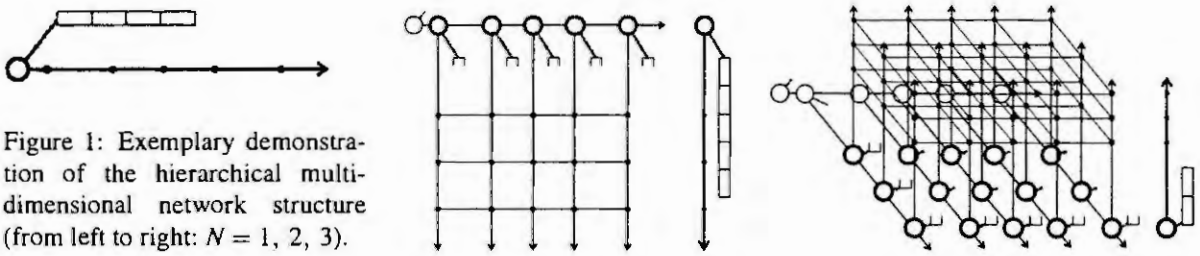


Figure 1: Exemplary demonstration of the hierarchical multi-dimensional network structure (from left to right: $N = 1, 2, 3$).

neuron positions. In contrast to the this common RBF-approach, the basis functions are developed independently in univariate dimensions in our strategy. Linear coefficients for the units of one dimension are calculated in a way, which fulfil the **Partition-of-unity**-condition: Based on this condition, the network output $\vec{W}(s)$ is calculated as weighted sum over all (participant) neurons K without normalisation.

$$\sum_{k=1}^{K} l_k(s) = 1 \quad \Rightarrow \quad \vec{W}(s) = \sum_{k=1}^{K} l_k(s)\vec{W}_k \tag{4}$$

Higher dimensional basis functions, which fulfil this condition again, are formed by taking the tensor product from the univariate ones. This enables a recursive calculation of the basis functions of all network layers for the determination of their merged output values $\vec{W}(\vec{s})$ (5). Because no multi-dimensional norm has to be calculated, this strategy is very effective. However, the produced basis functions do not have an ideal radial-symmetric form.

$$l_{\vec{k}}^{N}(\vec{s}) = \prod_{n=1}^{N} l_{k_n}(s_n) = l_{\vec{k}}^{N-1}(\vec{s}) \cdot l_{k_N}(s_N) \quad \Rightarrow \quad \vec{W}^{N}(\vec{s}) = \sum_{k_1=1}^{K_1} \cdots \sum_{k_N=1}^{K_N} l_{\vec{k}}^{N}(\vec{s})\vec{W}_{\vec{k}} \quad \text{with: } \vec{k} = [k_1, \ldots, k_N] \tag{5}$$

The B-Spline polynomials are applicable as such a basis function, because they fulfil the partition-of-unity-condition and have the following further properties [8]:

Recursion: $l_{i,p+1}(s) = \dfrac{s-\theta_i}{\theta_{i+p}-\theta_i} l_{i,p}(s) + \dfrac{\theta_{i+p+1}-s}{\theta_{i+p+1}-\theta_{i+1}} l_{i+1,p}(s) \qquad l_{i,1}(s) = \begin{cases} 1, & \text{if } s \in [\theta_i, \theta_{i+1}) \\ 0, & \text{otherwise} \end{cases}$

Positivity: $l_{i,K} \geq 0$ for all s **Local support:** $l_{i,K} = 0$ if $s \notin [\theta_i, \theta_{i+k}]$

The Neural Network Training

The model equation (3) is calculated after determination of the network output $\mathbf{ACGF}(\vec{\theta}) = \mathbf{W}(\vec{s})$ at the current robot position $\vec{s} = \vec{\theta}$ by subsumption of the output vectors $\vec{W}^N(\vec{s}) \in \mathbf{ACGF}(\vec{\theta})$ (5), which represent the elements of **ACGF** in all layers of the network according to Table 1. Because the input weights of the neurons are invariant as well as the form of the basis functions, only the output weights are subject of the network training. It is intended to perform this supervised training online, so that also an adaption to drifting parameters (e.g. changing friction) is enabled by a continuous learning process. Therefore, the learning of the correct parameters is managed by stochastic gradient descent utilising the recursive *Widrow-Hoff*-learning rule [7]. During the training, successive patterns consisting of a measured input vector $\vec{h}_{\theta i}$ and a teacher vector of torques measured too $\vec{\tau}_i$ are used. A new parameter matrix $\mathbf{W}^*$ is estimated, which minimises the squared error over all training patterns. All neurons $\vec{k}_r$, which contributed to the merged network output $\mathbf{W}(\vec{s})$, are adapted towards the individual estimation $\vec{W}^N \in \mathbf{W}^*$.

$$\vec{W}_{\vec{k}_r}^{new} = \vec{W}_{\vec{k}_r}^{old} + \eta \cdot l_{\vec{k}_r}^{N}(\vec{s}) \left(\vec{W}^N - \vec{W}_{\vec{k}_r}^{old} \right) \quad \text{with: } \mathbf{W}^* = \mathbf{W}(\vec{s}) + \frac{1}{\|\vec{h}_{\theta i}\|^2} \left(\vec{\tau}_i - \mathbf{W}(\vec{s}) \cdot \vec{h}_{\theta i} \right) \vec{h}_{\theta i}^{\mathrm{T}}, \tag{6}$$

The step size is determined by the individual coefficient of contribution $l_{\vec{k}_r}(\vec{s})$ and a learning rate η, which can be chosen in a range 0 to 1 due to the normalising term $1/\|\vec{h}_{\theta i}\|^2$ in (6).

Experimental Application and Results

As mentioned, an industrial robot SIEMENS manutec r2 is utilised for the experimental evaluation of the proposed identification strategy. This manipulator is additionally equipped with an open control system, implemented on a modular dSPACE-computer including multiple DSPs and ALPHA-processors. This system permits a real-time implementation of the whole neural architecture together the control of arbitrary robot trajectories.

The maximum network size, which can be realized with the available hardware, is a size of $9 \times 9 \times 9 \times 9$ neurons arranged in *equidistant* spacing (1 neuron per 20°). The required processing time is less than $1ms$, so that a synchronous execution with sample frequency of the trajectory controllers $1kHz$ is possible.

Unsynchronised PTP-movements of the axes are the system excitation during the network training, which has to comply with the following requirements: (i) All neurons of the network ought to pass through comparable numbers of training steps. By stochastic movement distances, which change direction only on the borders of the represented configuration space, all appearing joint positions are uniformly distributed. (ii) An uniformly excitation of all parameters can be verified as in common LS-identification by a low condition κ of the information matrix Φ. However, because of the multi-model approach and the open data set an analytic optimisation of κ is very complicated. Therefore, trajectories of minimal valued hitch r are chosen for the training (Fig. 2).

The presented architecture was tested with two different B-spline orders $K = 2$ and 3 at a period of $60min$ after beginning. The diagram of Fig. 3 shows, that the version of piecewise linear interpolation $K = 2$ achieved lower RMS-errors than the version of piecewise quadratic approximation $K = 3$. Apparently the occurring nonlinearities are not reproduced improved by quadratic approximation with the applied, unmodified low resolution.

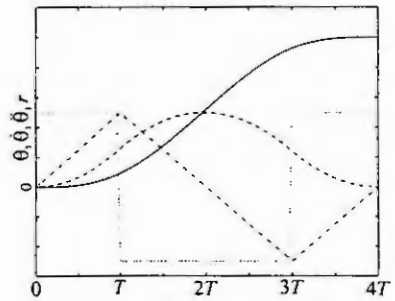


Figure 2: (left)
History of a trajectory with minimum hitch (solid: θ, dashed: $\dot{\theta}$, dashdot: $\ddot{\theta}$; dotted: r).

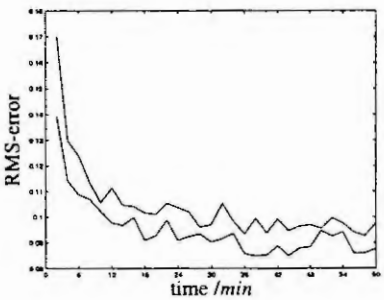


Figure 3: (right)
Comparison of the RMS-errors during network training (above: $K = 3$; below: $K = 2$).

A comparison of neural estimated and real measured torques resulted in significant errors only at signal peaks in the higher axes, but also qualitative differences in the lower ones certainly caused by the stronger couplings [4]. Therefore beyond extended training an experimental optimisation of the trajectories is required in future work.

Conclusion and Outlook

The presented architecture is able to model the dynamics of an industrial 6-joint robot online from experimental data. The neural technology solves by means of an optimal structural and architectural adjustment two problems: firstly, a flexible representation of nonlinear parameter dependences, and secondly, the experimental determination of all parameters from individual robot behaviour.

Future work is focused on synthesis of the total robot control system consisting of a neural network based linearising state feedback and individual joint controllers. An exact estimation of the unmodeled dynamics is very important to permit an adequate controller design, which is robust in respect of remaining model errors.

References

1. J. R. Beerhold. *Neuronale Radial-Basis-Funktion-Netze zur stabilen adaptiven Regelung von Gelenkarm-Robotern*. PhD thesis, Universität Wuppertal, 1995.
2. Craig, John J. *Introduction to Robotics Mechanics and Control*. Addison-Wesley, 2. edition, 1989.
3. M. Jansen. *Globale Modellbildung und garantiert stabile Regelung von Robotern mit strukturierten neuronalen Netzten*. PhD thesis, Universität Duisburg, 1995.
4. Krabbes, M. and Döschner, C. Modelling of robot dynamics based on multi-dimensional RBF-like neural network. In *IEEE Int. Conf. on Information, Intelligence, and Systems, Bethesda, USA*, IEEE Computer Society, 1999.
5. S. Miesbach. *Bahnführung von Robotern mit Neuronalen Netzen*. PhD thesis, TU München, 1995.
6. Poggio, T. and Girosi, F. *A theory of networks for approximation and learning*. AI Memo 1140. MIT, 1989.
7. Widrow, B. and Hoff, M.E. Adaptive switching circuits. In *IRE WESCON Convention Record, New York*, pages 96–104. IRE, 1960.
8. Zhang, J. and Knoll, A. Constructing fuzzy controllers with B-spline models-principles and applications. *Int. Journ. on Intelligent Systems*, 13(2/3):257–286, 1998.

ROBUST ADAPTIVE CONTROL OF ROBOTS BY MEANS OF STOCHASTIC OPTIMIZATION TECHNIQUES

K. Marti and A. Aurnhammer
Federal Armed Forces University
Aero-Space Engineering and Technology
D - 85577 Neubiberg/Munich, Germany

Abstract. The standard procedure in optimal control of robots, or more general dynamic systems, is replacing unknown parameters in the system equations, dynamic and kinemetic equation, resp., by nominal values. Due to the deviation between actual and nominal values, the resulting controls can be very sensitive to parameter variations. By introducing probability distributions for the modelling of the unknown parameters, stochastic parameter variations are incorporated in the optimal control process. Thus, more robust controls are obtained. In this paper especially the influence of different probability distributions in the modelling process of a Manutec r3 industrial robot with uncertain payload mass is studied.

Introduction

In adaptive optimal control of industrial or service robots the problem can be described at each stage j, after a time-path parameter transformation $s : [t_j, t_f^{(j)}] \mapsto [s_j, s_f]$ and a path parameter transformation $\tilde{s} : [s_j, s_f] \mapsto [\tilde{s}_0, \tilde{s}_f]$ onto a given fixed interval $[\tilde{s}_0, \tilde{s}_f]$, by means of a variational problem, depending on parameters like length of links, mass of links, payload mass, etc.. These parameters are not exactly known, but must be modelled by probability distributions, which are updated recursively whenever new information, obtained e.g. from parameter estimation algorithms, about the unknown parameters is available. This leads to a variational problem under stochastic disturbances. Using stochastic optimization techniques, this stochastic variational problem finally is replaced by a deterministic substitute problem of the following type [2]:

$$\min_{\tilde{\beta}(\cdot),\tilde{q}_e(\cdot)} \int_{\tilde{s}_0}^{\tilde{s}_f} \mathcal{E}(f_0(s, \tilde{q}_e(\tilde{s}), \tilde{q}_e'(\tilde{s})\tfrac{\tilde{s}_f-\tilde{s}_0}{s_f-s_j}, \tilde{q}_e''(\tilde{s})\left(\tfrac{\tilde{s}_f-\tilde{s}_0}{s_f-s_j}\right)^2, \tilde{\beta}(\tilde{s}), \tilde{\beta}'(\tilde{s})\tfrac{\tilde{s}_f-\tilde{s}_0}{s_f-s_j}, p_J) \mid \mathcal{A}_{t_j})\, ds \quad \text{(1a)}$$

subject to

$$\tilde{\beta}(\tilde{s}_0) = \beta_j,\ \tilde{\beta}(\tilde{s}_f) = 0, \quad \text{(1b)}$$

$$P(\tau_{min,i} \le \tilde{a}_i\tilde{\beta}'(\tilde{s})\tfrac{\tilde{s}_f-\tilde{s}_0}{s_f-s_j} + \tilde{b}_i\tilde{\beta}(\tilde{s}) + \tilde{c}_i \le \tau_{max,i} \mid \mathcal{A}_{t_j}) \ge \alpha,\ \tilde{s}_0 \le \tilde{s} \le \tilde{s}_f,\ i = 1, 2, \ldots, n, \quad \text{(1c)}$$

$$P(q_{min} \le \tilde{q}_e(\tilde{s}) \le q_{max} \mid \mathcal{A}_{t_j}) \ge \alpha,\ \tilde{s}_0 \le \tilde{s} \le \tilde{s}_f, \quad \text{(1d)}$$

$$P(\dot{q}_{min} \le \tilde{q}_e'(\tilde{s})\tfrac{\tilde{s}_f-\tilde{s}_0}{s_f-s_j}\sqrt{\tilde{\beta}(\tilde{s})} \le \dot{q}_{max} \mid \mathcal{A}_{t_j}) \ge \alpha,\ \tilde{s}_0 \le \tilde{s} \le \tilde{s}_f, \quad \text{(1e)}$$

$$\tilde{\beta}(\tilde{s}) \ge 0,\ \tilde{s}_0 \le \tilde{s} \le \tilde{s}_f, \quad \text{(1f)}$$

$$\tilde{q}_e(\tilde{s}_0) = q_j,\ \tilde{q}_e(\tilde{s}_f) = q_f. \quad \text{(1g)}$$

Here, the σ-Algebra $\mathcal{A}_{t_j}$ represents the information available up to time t_j, $q = \tilde{q}_e(\tilde{s})$ is the vector of configuration or robot coordinates, $\beta = \tilde{\beta}(\tilde{s})$ the velocity profile and $p := (p_D, p_K, p_J)^T$ the vector of unknown dynamic, kinematic and objective model parameters. The coefficients $\tilde{a}_i = \tilde{a}_i(\tilde{q}_e, \tilde{q}_e', s_j, p_D)$, $\tilde{b}_i = \tilde{b}_i(\tilde{q}_e, \tilde{q}_e', \tilde{q}_e'', s_j, p_D)$ and $\tilde{c}_i = \tilde{c}_i(\tilde{q}_e, s_j, p_D)$, $i = 1, \ldots, n$, are obtained from the dynamic equation of the robot [3], where n is the number of degrees of freedom equivalent to the number of joints. Moreover, (1b) are the initial and terminal values for the velocity profile $\beta = \tilde{\beta}(\tilde{s})$, (1c) represents control constraints, which restrict the forces and moments in the robot joints, and (1d,e) are position and velocity restrictions for the robot path in configuration space. Finally, α is the prescribed reliability, and the objective function (1a) can describe different optimization criteria like minimum-time or minimum energy consumption.

Hence, the solution depends on the available information $\mathcal{A}_{t_j}$ at stage j and on the chosen probability distribution that models the initial uncertainty about the parameter p.

Solution techniques

Approximating the unknown functions $\tilde{\beta}(\tilde{s})$ and $\tilde{q}_e(\tilde{s})$ by linear combinations of known basis functions, e.g. B-splines, and demanding that the restrictions that depend on the path parameter $\tilde{s}$, $\tilde{s}_0 \leq \tilde{s} \leq \tilde{s}_f$, are fullfilled in certain knots $\tilde{s}_i$, $i = 1, \ldots, m$, problem (1a-g) is reduced to a ordinary nonlinear parameter optimization problem. Another possible solution technique is applying calculus of variation techniques that provide necessary and somtimes sufficient conditions in terms of differential equations [3].

Once optimal solutions $q_e^{(j)}, \beta^{(j)}$ are derived from (1a-g), feedforward controls are calculated using the inverse dynamics approach:

$$\tilde{a}_i(q_e^{(j)}, q_e^{(j)'}, s_j, p_D) \cdot \beta^{(j)'}(\tilde{s}) \cdot \frac{\tilde{s}_f - \tilde{s}_0}{s_f - s_j} + \tilde{b}_i(q_e^{(j)}, q_e^{(j)'}, q_e^{(j)''}, s_j, p_D) \cdot \beta^{(j)}(\tilde{s}) + \tilde{c}_i(q_e^{(j)}, s_j, p_D) = \tau_i, i = 1, 2, \ldots, n. \tag{2}$$

Numerical example: Manutec r3

The industrial robot Manutec r3 has 6 revolute joints, where for path planning [4] only the first three joints are taken into account.

Let without loss of generality $[s_0, s_f] = [\tilde{s}_0, \tilde{s}_f]$ and consider the following time-optimal point-to-point problem under position and velocity constraints at the initial stage $j = 0$:

$$q_e(s_0) = \begin{pmatrix} -2.4 \\ 1.2 \\ 0.6 \end{pmatrix}, \; q_e(s_f) = \begin{pmatrix} -1.3 \\ 0.2 \\ -1.0 \end{pmatrix}, \tag{3}$$

where the following restrictions have to be fullfilled [4]:

$$P(-7.5 \leq \tau_i(s) \leq 7.5 \mid \mathcal{A}_{t_0}) \geq \alpha, \; i = 1, 2, 3, \tag{4}$$

$$-2.97 \leq q_1(s) \leq 2.97, \tag{5a}$$

$$-2.01 \leq q_2(s) \leq 2.01, \tag{5b}$$

$$-2.86 \leq q_3(s) \leq 2.86, \tag{5c}$$

$$-3.0 \leq q_1'(s)\sqrt{\beta(s)} \leq 3.0, \tag{6a}$$

$$-1.5 \leq q_2'(s)\sqrt{\beta(s)} \leq 1.5, \tag{6b}$$

$$-5.2 \leq q_3'(s)\sqrt{\beta(s)} \leq 5.2, \tag{6c}$$

with $\alpha = 0.99$. Using the software OSPP [1] developed at the Federal Armed Forces University, we study three possible probability distributions for a single random model parameter: the payload mass m_l. Additionally, the solution of the deterministic problem is given, where the payload is assumed to be exactly known with a mass of 5 kg. Furthermore, in the cases of a stochastic payload we assume that the payload has expectation $\mathcal{E}(m_l) = 5$ and variance $Var(m_l) = 25$. Hence, for the uniform, exponential and gaussian distribution we get the following results, where we took the inverse of the time-path parameter transformation to plot the results in time domain:

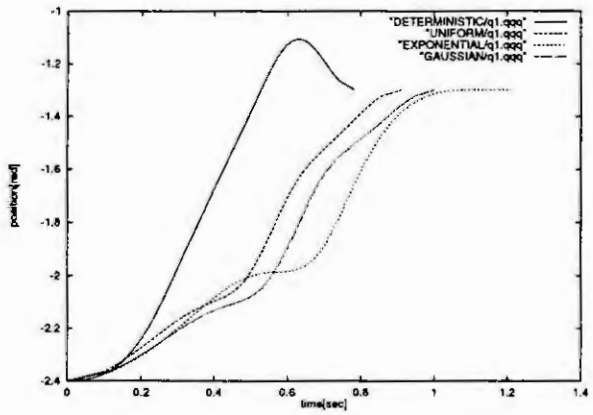


Figure 1: Position $q_1(t)$ of the first joint.

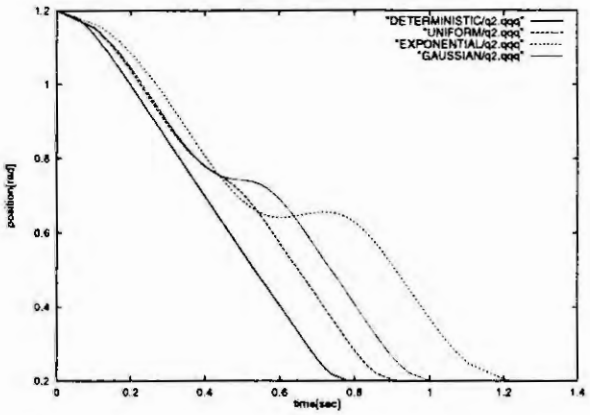


Figure 2: Position $q_2(t)$ of the second joint.

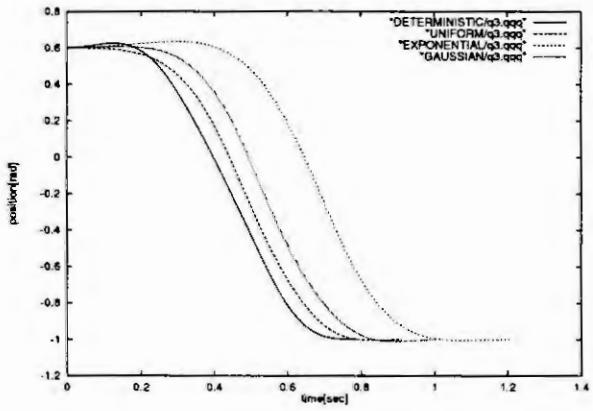


Figure 3: Position $q_3(t)$ of the third joint.

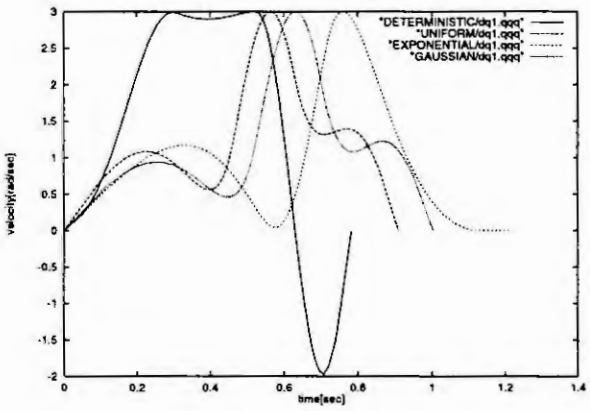


Figure 4: Velocity $\dot{q}_1(t)$ of the first joint.

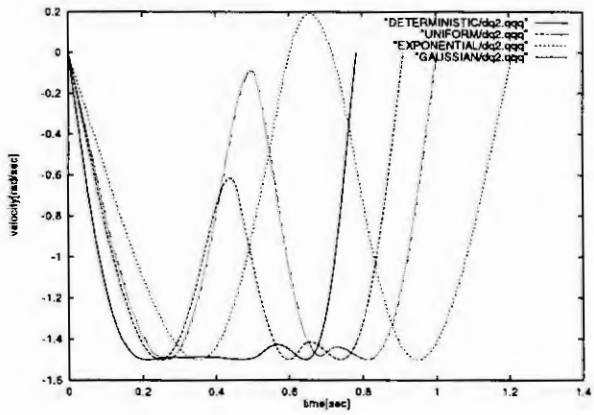


Figure 5: Velocity $\dot{q}_2(t)$ of the second joint.

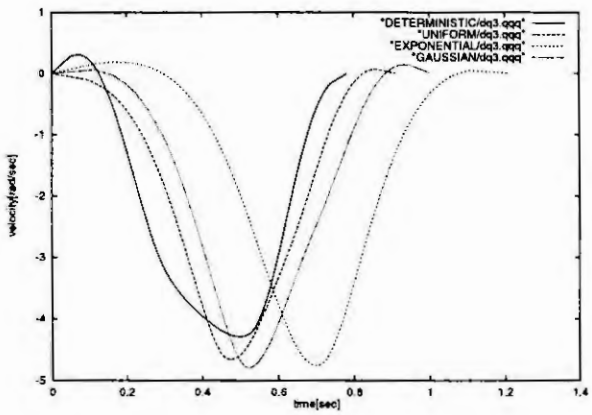


Figure 6: Velocity $\dot{q}_3(t)$ of the third joint.

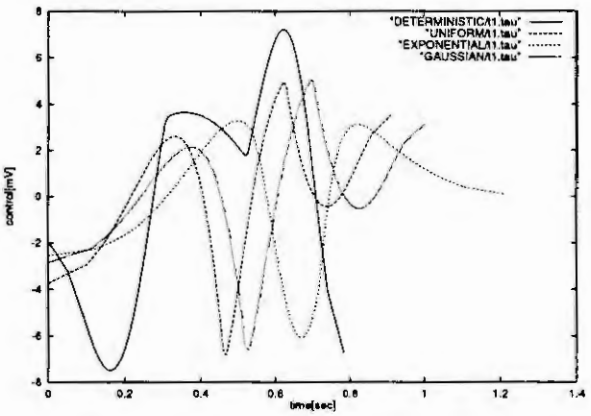


Figure 7: Control input $\tau_1(t)$ of the first joint.

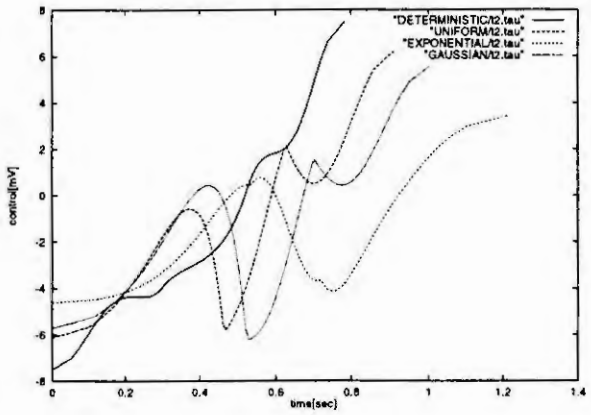


Figure 8: Control input $\tau_2(t)$ of the second joint.

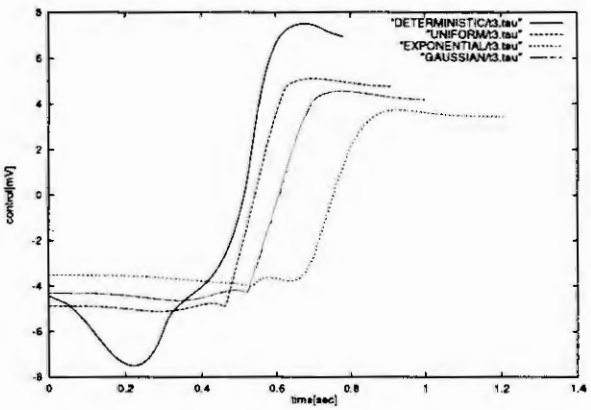


Figure 9: Control input $\tau_3(t)$ of the third joint.

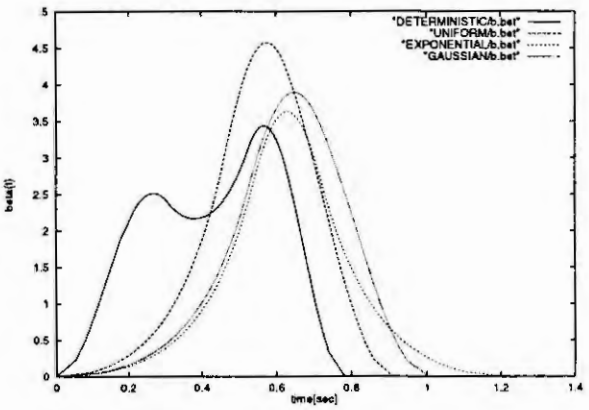


Figure 10: Velocity profile $\beta(t)$

Obviously, in case of an exponential distribution the most cautious controls are obtained, see Fig. 7-9. Comparing the support of the uniform and exponential distribution, we have that:

$$supp_{uni}(m_l) = [5 \cdot (1 - \sqrt{3}), 5 \cdot (1 + \sqrt{3})], \; supp_{exp}(m_l) = \mathcal{R}^+. \tag{7}$$

Hence, in case of the exponential distribution a payload mass greater than $5 \cdot (1 + \sqrt{3})$ contributes with positive probability to the calculation of the optimal control. Additionally, examing the distribution function $F^{exp}_{m_l}(z) = P(m_l \leq z)$, we have still a probability for the payload to be greater than $5 \cdot (1 + \sqrt{3})$ of about 0.065. Thus, the exponential distribution incorporates a much wider range of possible payload masses (assumed that expectation and variance are equal for both distributions). Similar relations between the other distributions can be obtained.

References

[1] Aurnhammer, A.; Marti, K.: *OSPP Fortran-Program for Optimal Stochastic Path Planning of Robots*, to appear

[2] Marti, K.: *Adaptive Stochastic Path Planning and Control (ASPPC) for Robots*, to appear

[3] Qu, S.: *Optimale Bahnplanung für Roboter unter Berücksichtigung stochastischer Parameterschwankungen*, VDI Verlag, Düsseldorf, 1995

[4] Türk, M.: *Zur Modellierung der Dynamik von Robotern mit rotatorischen Gelenken*, VDI-Verlag, Düsseldorf, 1990

Modeling and control of a flexible-link manipulator

Dadi Hisseine, Boris Lohmann
University of Bremen, Institute of Automation Technology
Kufsteiner Strasse, FB1 / NW1, 28359 Bremen, Germany
e-mail: hisseine @iat.uni-bremen.de, bl@iat.uni-bremen.de

Abstract. By a suitable description of the considered system dynamics, different tracking approaches for controlling single link-flexible manipulators are presented in this paper. First, on the basis of a distributed parameter model, stabilizing feedback with additional use of nonlinear strain feedback signals, for counteracting destabilizing disturbances following the open loop control [1], is introduced. Second, on the basis of a discrete nonlinear flexible model, a robust nonlinear control law is derived using continuous sliding mode techniques. The performances of the presented control strategies are demonstrated by simulations and experiments carried out with the flexible robot arm of our Institute.

1. Introduction

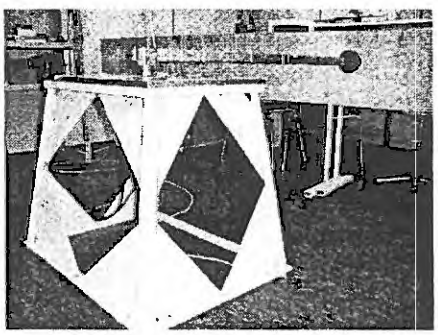
The topic "Dynamics and control of flexible-link robots " represents a pretentious and challenging problem. In order to be able to counteract the undesirable effects of flexibility, advanced robot control techniques should be investigated on the basis of a more complete dynamic. For control design, different approaches exist for the description of dynamics of flexible-link manipulators. Exact solution approaches as in [1], which deal with the partial differential equations of system dynamics, are available as well as approximation approaches such assumed modes methods [2]. Due to the inherent unstable zero dynamics behavior of the end-effector position of flexible-link robots, the simultaneous achievement of high level performance and robustness is not straightforward. A variety of control policies have been used by researchers to control flexible manipulators. In this paper we present, by a suitable description of the considered system dynamics, different control approaches.

First, we present a controller, which is obtained from the distributed parameter model of the flexible robot arm. In an algebraic framework using Mikusinski operational calculus [4], the authors in [1] have investigated the trajectory tracking problem for a flexible robot arm. Note that this methodology leads to exact solutions without integration of any differential equation. In order to improve the performance of the passivity based concept in [1], we examine here the additional use of nonlinear strain feedback signals. Performance improvements relative to the trajectory tracking, which are demonstrated by means of simulations and experiments, are obtained.

Second, we propose, on the basis of a discrete nonlinear flexible model, a robust control law, which is derived using a sliding mode strategy. Unlike distinguishing properties such robustness to modeling uncertainties and disturbances, the use of switched control is even associated with negative effects such as chattering phenomenon [5]. In order to overcome the undesired control chattering (due to the high frequency switching), the considered approach satisfies the sliding condition using a strategy, which avoids the discontinuity in the classical sliding mode techniques. Our approach is based on the continuous sliding mode control, which is illustrated in [6]. Two alternative system outputs (tip postion or joint variable) can be considered for trajectory tracking. By a suitable choice of the sliding surface, we show here, that the tracking problem can be solved by considering the joint variable as the minimum phase system output. Simulations studies with a considerable amount of parametric uncertainties are to demonstrate the robustness of the proposed tracking controller.

2. Modeling

The single link flexible robot arm under consideration, presented in figure 1, is modelled as an Euler-Bernoulli beam. The motion occurs only in the horizontal plane and the arm does not undergo torsional deformations.



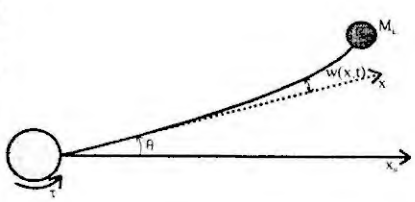


Figure 1: Flexible-link robot arm

The motion of a point along the beam, is given by $y(x,t) = x \cdot \theta(t) + w(x,t)$ (1)

Using Lagrange's Formulation or Hamilton's Principle, we obtain a fourth-order partial differential equation:

$$EI \cdot \frac{\partial^4 y(x,t))}{\partial x^4} + \rho A \cdot \frac{\partial^2 y(x,t)}{\partial t^2} = 0\,, \quad \text{with 4 boundary conditions as given in [3],} \tag{2}$$

This partial differential equation of the system dynamics can be either directly solved (exact solution approach such in [1]) or using approximation approaches such as assumed modes methods (see section 4).

3. Control based on an exact solution approach

3.1 Open loop control

In an algebraic framework using Mikusinski operational calculus [4], the authors in [1] have investigated the trajectory tracking control problem of a flexible robot arm presented by an Euler-Bernoulli beam. The methodology presented in [1] leads to an exact solution without integration of any differential equation.

In order to transform the flexible robot arm as Mikusinski system, we set $\tilde{t} = \sqrt{\frac{\rho A}{EI}} L^2 \cdot t$, $\tilde{x} = Lx$ and obtain from (2): $\frac{\partial^2 y(\tilde{x},\tilde{t})}{\partial \tilde{t}^2} = -\frac{\partial^4 y(\tilde{x},\tilde{t})}{\partial \tilde{x}^4}$. With the initial conditions $y(x,0) = 0$ and $\dot{y}(x,0) = 0$, the transformation of this linear partial differential equation using Mikusinski operational calculus yields: $s^2 \hat{y}(\tilde{x},s) = -\hat{y}^{(4)}(\tilde{x},s)$, where $\hat{y}$ denotes the operational function corresponding to y. Let y_d be the tip position (system output variable). From the boundary conditions and after some calculations, the input control torque $\tau(t) = u(t)$ can be, with the Mikusinski derivative operator s, formulated as $\hat{u}\left(s, \hat{y}_d\right)$.

The back transformation to time scale requires some restrictions relative to the output function $y_d(t)$, which must be an analytical function involving an infinite number of derivatives. It has been shown in [1], that if $y_d(t)$ appertains to a specific class of functions, the following series, which corresponds to the **input control torque** $u(t)$, is absolutely convergent:

$$u(t) = \frac{J_m}{L\alpha^2}\left[1 + \sum_{n=0}^{\infty} \frac{2^{2n+1}}{(4n+4)!} \psi_n^1\left(\frac{d}{dt}\right)\right] \cdot \ddot{y}_d(t) - \frac{EI}{L^2}\left[\sum_{n=0}^{\infty} \frac{2^{2n+1}}{(4n+4)!} \psi_n^2\left(\frac{d}{dt}\right)\right] \cdot \ddot{y}_d(t) \tag{3}$$

where $\psi_n^1\left(\frac{d}{dt}\right) = \left((1+\lambda\mu)\frac{d^2}{dt^2} + (4n+4)\left(\mu + \frac{4n+3}{2}\lambda\right)\right)\frac{d^{2n+2}}{dt^{2n+2}}$ and

$\psi_n^2\left(\frac{d}{dt}\right) = \left((4n+4)\left(\frac{1}{2} + \frac{\lambda\mu}{2}\right)\frac{d^2}{dt^2} + (4n+3)\left(\mu + \frac{(4n+1)(4n+2)}{2}\lambda\right)\right)\frac{d^{2n}}{dt^{2n}}$, with $\lambda = \frac{J_L}{\rho A L^3}$ and $\mu = \frac{M_L}{\rho A L}$.

3.2 Passivity based control with additional nonlinear strain feedback

In order to counteract destabilizing disturbances and to achieve asymptotic stability following the open loop control (5), stabilizing feedback is necessary. By assigning a reference trajectory denoted by (θ_d, $\dot{\theta}_d$), a passivity based feedback can be introduced:

$$u = \tau_d(t) - K_p \cdot \left(\theta - \theta_d\right) + K_v \cdot \left(\dot{\theta} - \dot{\theta}_d\right), \text{ where } \tau_d(t) = -\frac{EI}{L^2}\frac{\partial^2 y\left(0, \frac{t}{\alpha}\right)}{\partial x^2} \text{ is the reference torque,} \tag{4}$$

Following the preceding regulation, we introduce, in addition to (4), a nonlinear strain feedback [7] for performance improvements:

$$u = -K_p \cdot \left(\theta - \theta_d\right) + K_v \cdot \left(\dot{\theta} - \dot{\theta}_d\right) - \left(\sum_{i=1}^{D} K_f \cdot w''(x_s,t)\int_0^t \dot{\theta}(\xi) w''(x_s,\xi)\,d\xi - \tau_d(t)\right) \tag{5}$$

where K_p, K_v and K_f are positive parameters and $x_{si} \in [0, L]$ is the location of the *ith*-strain gauge and D represents the number of strain gauges on the flexible robot arm.

3.3 Experimental results

The physical parameters of the considered flexible robot system are: $L = 1.155m$, $\rho A = 2660 \times 3.2\,10^{-4} Kg/m$, $EI = 7\,10^{10} \times 1.71\,10^{-9} N m^2$, $J_m = 3.2\,10^{-3} Kg\,m^2$, $M_L = 5Kg$, $J_L = 7.96\,10^{-3} Kg\,m^2$

Let y_d be the desired reference trajectory: $y_d(t) = K \cdot (h * g)(t)$, where K is constant, $h(t)$ represents the step function and $g(t) = e^{-\left(\frac{t}{T}(1-\frac{t}{T})\right)^{-\upsilon}}$, with $\upsilon = 1.1$, $t \in]0, T[$, the so-called Gevrey-Roumieu function.

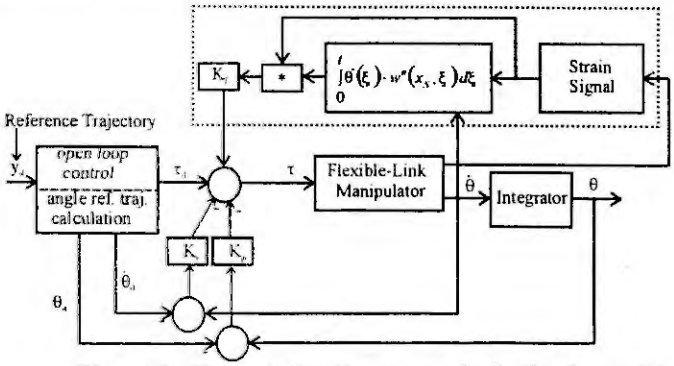


Figure 2: Control plan (by one strain feedback signal)

Compared to the design in [1], we can ascertain performance improvement by simulations and experiments as shown in the following figures 3 and 4 (and for more details, see [3]).

Passivity based control with additional nonlinear strain feedback: (···) *simulated* and (—) *measured values*

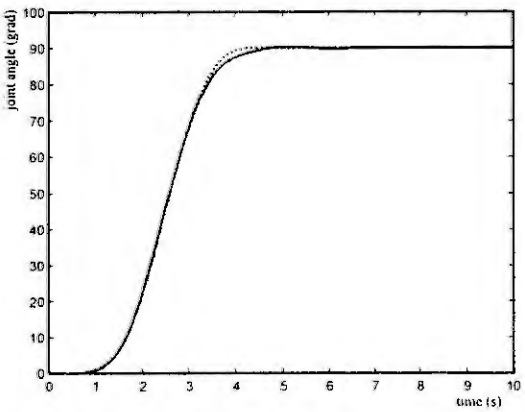


Figure 3 Angular motion (grad)

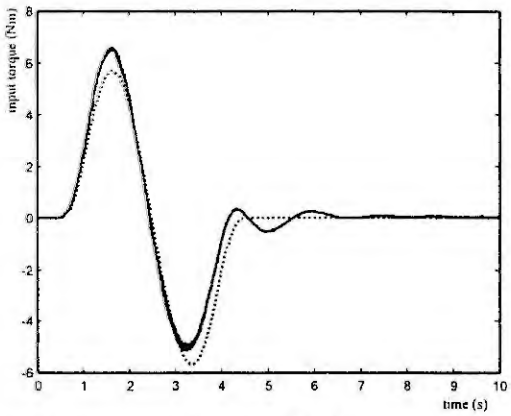


Figure 4: Input control torque (Nm)

4. Robust nonlinear tracking control

4.1 Model description in state space formulation

In the following, the flexible-link manipulator under consideration (see figure 1), is now modeled by an approximation approach using assumed mode methods. By consideration of a finite number m of modal terms, the system dynamics are derived using Lagrange Formulation (see [3]):

$$\mathbf{M}(\boldsymbol{\delta}) \begin{pmatrix} \ddot{\theta} \\ \ddot{\boldsymbol{\delta}} \end{pmatrix} = \begin{pmatrix} -C_\theta(\dot{\theta}, \boldsymbol{\delta}, \dot{\boldsymbol{\delta}}) + u \\ -\mathbf{C}_\delta(\dot{\theta}, \boldsymbol{\delta}) - \mathbf{F} \cdot \dot{\boldsymbol{\delta}} - \mathbf{K} \cdot \boldsymbol{\delta} \end{pmatrix}, \tag{6}$$

where $\mathbf{M}(\boldsymbol{\delta})$ the positive definite symmetric inertia matrix, θ the joint variable, $\boldsymbol{\delta} = [\delta_1 \quad \cdots \quad \delta_m]^T$ the vector of modal amplitudes, C_θ and $\mathbf{C}_\delta$ coriolis and centrifugal terms respectively, $\mathbf{F}$ the structural damping matrix, $\mathbf{K}$ the stiffness matrix and u the input torque at the joint.

For the purpose of design, we will transform the above system into the normal form, using differential geometric methods. By choosing the motor position angle as output variable, the precedent flexible-link robot system has a relative degree of 2, (i.e. there exists an unobservable subsystem, the so-called zeros dynamics). It can be easily shown, that the zeros dynamics associated with this output is stable, i.e. the above system is minimal phase. With $\mathbf{x} = (\theta \quad \dot{\theta})^T$, $\boldsymbol{\eta} = (\boldsymbol{\delta} \quad \dot{\boldsymbol{\delta}})^T$ and the joint angle as the alternative system output $y = x_1$, the system equation (6) can be represented into the so-called Byrnes–Isidori normal form [3]:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \boldsymbol{\eta}) + \mathbf{g}(\mathbf{x}, \boldsymbol{\eta}) \cdot u \\ \dot{\boldsymbol{\eta}} &= \boldsymbol{\Psi}(\boldsymbol{\eta}) \end{aligned} \tag{7}$$

4.2 Tracking Control using continuous sliding mode approach

An input control torque , which is capable of exactly reproducing a given trajectory $\mathbf{x}_d(t)$, can be derived by means of continuous sliding mode techniques. The sliding mode control forces the system trajectory to reach and stay on the sliding surface. We select the sliding surface as follows: $s = \mathbf{c}^T\mathbf{e}$, where $\mathbf{e} = \mathbf{x} - \mathbf{x}_d$ represents the tracking error and $\mathbf{c} = \begin{pmatrix} c_1 & 1 \end{pmatrix}^T$ with $c_1 > 0$. Let $V(\mathbf{x})$ be a Lyapunov function : $V(\mathbf{x}) = \frac{1}{2}s^2$ (8)

The sliding condition is fulfilled by the choice of the control law

$$u = -\left(\mathbf{c}^T\mathbf{g}\right)^{-1}\mathbf{c}^T\mathbf{f} + \left(\mathbf{c}^T\mathbf{g}\right)^{-1}\mathbf{c}^T\dot{\mathbf{x}}_d - \alpha \cdot s \cdot \mathbf{c}^T\mathbf{g} = -\varphi(\mathbf{x}) - \alpha \cdot \sigma(\mathbf{x}),\ \alpha > 0 \tag{9}$$

such that the time derivative of the Lyapunov function is negative definite, i.e. $\dot{V}(\mathbf{x}) = -\alpha \cdot \left(s\,\mathbf{c}^T\mathbf{g}\right)^2 < 0$, $\forall\, \mathbf{x} \neq \mathbf{x}_R$, where $\mathbf{x}_R$ represents the steady state.

Robustness to modeling uncertainties: In order to counteract the inherent chattering phenomen in classical sliding mode control, Slotine [5] have introduced the concept of „boundary layer". According to this approach and in order to deal explicitly with parameter variations and disturbances, the concept of boundary layer equivalence for the continuous sliding mode control is illustrated in [6]. Here, the boundary layer is imposed on the switching curve $s \cdot \mathbf{c}^T\mathbf{g} = 0$ instead of directly on the sliding mode $s(\mathbf{x}) = 0$. In the presence of bounded modeling errors in $\mathbf{f}(\mathbf{x})$ and $\mathbf{g}(\mathbf{x})$ of equation (7), equation (9) yields:

$$u_p = -\frac{1}{2}\left(\sup\varphi(\mathbf{x}) + \inf\varphi(\mathbf{x})\right) - \frac{\sigma(\mathbf{x})}{2\lambda}\left(2\alpha + \sup\varphi(\mathbf{x}) - \inf\varphi(\mathbf{x})\right), \tag{10}$$

where λ represents the width of the boundary layer.

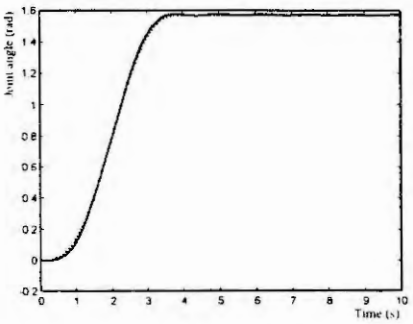


Figure 5: Joint angle (rad)

Simulations results: The sliding mode control law (10) has been tested by means of simulations using a two-mode nonlinear model (6) of the single link flexible robot arm. We choose the reference trajectory denoted by $\mathbf{x}_d(t) = \begin{pmatrix} \theta_d(t) & \dot{\theta}_d(t) \end{pmatrix}^T$ as an angular motion from $\theta_d(0) = 0$ to $\theta_d(T) = \frac{\pi}{2}$, with the velocity profile $\dot{\theta}_d(t) = \frac{\pi}{2T}\left(1 - \cos\left(\frac{2\pi}{T}t\right)\right)$. Simulations results, which are shown in figures (5-6), demonstrate the effectivness of the proposed approach.

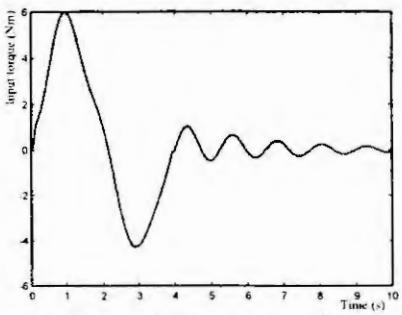


Figure 6: Input torque (Nm)

5. Conclusion

Different approaches for the trajectory control of the robot have been investigated in theory, simulation and experiments. By introducing an additional nonlinear strain feedback to the passivity based control, performance of the closed loop system is improved. A nonlinear robust control strategy (based on the concept of continuous sliding mode) was proposed and tested by simulation studies. The presented methodologies form in each case an efficient and robust tool for solving the trajectory tracking problem for single link-flexible manipulators.

6. References

[1] Y. Aoustin, M. Fliess, H. Mounier, P. Rouchon and J. Rudolph, Theory and practice in the motion planning and control of a flexible robot arm using Mikusinski operators. In *Proc. 4th Symp. Robotics and Control*, 1997.

[2] A. De Luca, Trajectory control of flexible manipulators. In: B. Siciliano and K.P.Valvanis (Eds), *Control Problems in Robotics and automation* (London: Springer-Verlag, 1998).

[3] D. Hisseine, B. Lohmann and A. Kuczynski, Two control approaches for a flexible-link manipulator. In *Proc. IASTED Int. Conf., Robotics and Automation, RA '99*, Santa Barbara, USA, 1999.

[4] J. Mikusinski, *Operational Calculus*, 2nd ed., Vol. I. (Oxford: Pergamon Press, 1987).

[5] J.J. Slotine, Sliding controller design for nonlinear systems , *Int. J. Control*, 40, 1984, 421-434.

[6] F. Zhou and D.G. Fisher, continuous sliding mode control, *Int. J. Control*, 55, 1992, 313-327.

[7] S.S. Ge, T.H. Lee and G. Zhu, Improving Regulation of a single link flexible manipulator with strain Feedback, *IEEE Trans. Robotics and Automation*, 14, 1998, 179-185.

MODELLING AND SIMULATION OF LINK AND JOINT FLEXURE IN A LIGHTWEIGHT ROBOT MANIPULATOR

Alan S. Morris
University of Sheffield, Sheffield S1 3JD, UK.

Abstract. The paper is concerned with a robot manipulator having two lightweight links. This is modelled as a composite system of two flexible links and four rigid ones. Rigid link models have been widely reported and hence this paper is only concerned with the modelling of the flexible two link system which can be subsequently integrated with the rigid link part of the system. The approach taken is to develop models of a single flexible link and joint and then consider the relevant dynamic coupling parameters when the single joint-link model is extended to a two joint-link system.

1 Introduction

Much interest has been shown recently in lightweight, flexible robot manipulators made from composite materials, since these avoid the large and problematical inertia forces associated with traditional heavy, large-section, rigid-link manipulators. However, the introduction of flexibility, and the consequent tendency of the links to oscillate during motion, creates a control problem for which a very accurate model of the flexure mechanisms is required. Such a model must not only describe deflection and oscillation of flexible links but also describe flexure and oscillation in the actuated joints that connect the links together.

Robot manipulators that are currently in use commonly have six degrees of freedom. However, analysis reveals that it is only the motion of two of the links in a typical manipulator, as shown in figure 1, that cause the problematical inertia forces. In consequence, only these two links need to be lightweight and hence flexible. Thus, if the motion of these two flexible links can be modelled accurately, this sub-model can be combined with a model of the dynamics of the other four degrees-of-freedom modelled as a rigid-link system in order to obtain a composite model of the six-degree-of-freedom system.

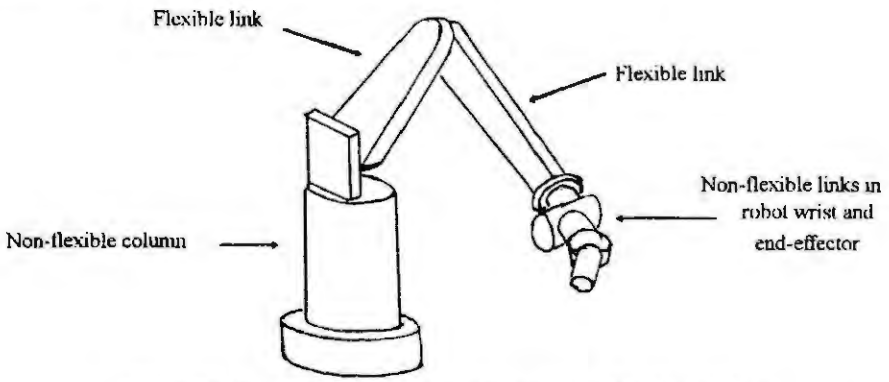


Fig 1: Robot system with two lightweight, flexible links

This paper therefore describes the modelling of link and joint motion in a two-flexible-link system, after which the addition of a model of the other rigid links in the manipulator is relatively simple. The paper concentrates particularly on optimising model accuracy and computational efficiency. The starting point for such a model is the large volume of work reported previously on modelling the deflection of single links. However, all of this previous work makes approximations that involve limiting motion to small displacements and omitting consideration of shear forces. Such approximations are valid for a single link but they are invalid for multiple links, where dynamic coupling exists between the links and modelling errors are cumulative from one link to the next. Previous work at Sheffield has resulted in the development of a model that avoids small-displacement approximations, includes the effect of shear forces and also includes the dynamic coupling between links in a two-link system, thereby providing a model of a two-link system with much improved accuracy. This will be briefly reviewed before the more recent addition of flexible joint modelling is described. Simulation of the performance of the combined flexible link and joint model will then be described in the paper presented.

Finally, the combination of the two-flexible-link model developed with a model of the other four rigid links in a typical six-degree-of-freedom robot manipulator will be explained.

2 Modelling of single flexible link

Whilst there has been a large volume of work reported on modelling single flexible links, these generally make approximations that render the model invalid for use in a multi-link system. Such approximations usually take the form of assuming that flexural deflections of the link are small and that shear deformation is negligible. When a single-link model is to form part of a multi-link system, inter-link coupling becomes very important, and therefore very accurate modelling of the end-tip slope and velocity as well as its position is required. This, in turn, requires that account is taken of shear deformation effects and that small-angle approximations are not made.

The static slope du/dx and deflection u at a distance x from the actuated end along a flexible link with distributed mass m and end-tip load m_t, neglecting the shear deformation effect and assuming small deflections, are given by [1]:

$$\frac{du}{dx} = \frac{du_m}{dx} + \frac{du_{mt}}{dx} = -\frac{g}{2EIl}\left((m+2m_t)\frac{l^2x}{2} - (m+m_t)\frac{lx^2}{3} + m\frac{x^3}{6}\right) \tag{1}$$

$$u = u_m + u_{mt} = -\frac{g}{2EIl}\left((m+2m_t)\frac{l^2x^2}{2} - (m+m_t)\frac{lx^3}{3} + m\frac{x^4}{12}\right) \tag{2}$$

where l is the length of the beam, EI is the flexural stiffness of the beam and g is the acceleration due to gravity. The maximum static slope and deflection of the flexible link occur at the free end, where $x = l$, i.e.

$$\frac{du_{\max}}{dx} = -\frac{l^2g}{2EI}(m/3+m_t) \quad ; \quad u_{\max} = -\frac{l^3g}{2EI}(m/4+2m_t/3) \tag{3}$$

As these equations only assume small deflections, correction is required when the deflection is not small. The deflection of the link end-tip is calculated above on the assumption that the end-tip moves vertically downwards instead of in a circular arc. This is clearly only valid if the magnitude of flexure is low. This condition is unlikely to be satisfied in typical industrial flexible manipulator links. Previous work has shown that the case of large magnitude flexure can be handled by adding a correction factor to the basic equations. This is calculated by considering the link as a body composed of n equal sections and applying finite element analysis. The corrected co-ordinates of the link end-tip are then given by [1]:

$$x_e = l - s \quad \text{and} \quad y_e = u(l-s) \tag{4}$$

where:

$$u(l-s-w_n) = \frac{w_n}{v_n}u(l-s)$$

$$s = \sum_{i=1}^{n-1} w_i \quad ; \quad v_n = L - l/n \quad ; \quad w_n = \frac{v_n l}{nL} \quad ; \quad L = \sqrt{[u(l-s) - u((n-1)l/n - s)]^2 + [l/n]^2}$$

The equation of motion of a flexible link is given in [1] as:

$$\rho\frac{\partial^2 u(x,t)}{\partial t^2} = -\frac{\partial^2}{\partial x^2}\left(EI\frac{\partial^2 u(x,t)}{\partial x^2}\right) \quad ; \quad u(x,t) = \phi(x)q(t) \tag{5}$$

where ρ is the mass per unit length of the link, $u(x,t)$ is the deflection of the link, $\varphi(x)$ is the assumed mode shape function and $q(t)$ is the modal function. Assuming that EI is a constant, equation (5) can be written as:

$$\frac{1}{q(t)}\frac{d^2q(t)}{dt^2} = -\frac{EI}{\rho}\frac{1}{\phi(x)}\frac{d^4\phi(t)}{dx^4} \tag{6}$$

which leads to the two following differential equations:

$$\frac{d^4\phi(x)}{dx^4} - \beta^4\phi(x) = 0 \quad ; \quad \frac{d^2q(t)}{dt^2} + \omega^2 q(t) = 0 \tag{7}$$

where ω is a constant and $\beta^4 = \rho\omega^2/EI$. The solution is given in [1] as:

$$\phi_i(x) = C_i(\cos\beta_i x - \cosh\beta_i x) + (\sin\beta_i x - \sinh\beta_i x) \quad ; \quad q_i(t) = A_i\cos\omega_i t + B_i\sin\omega_i t \tag{8}$$

where A_i, B_i, C_i, and ω_i are constants, i denotes the number of modes of vibration. The deflection is then given by:

$$u(x,t) = \sum_{i=1}^{\infty} \phi_i(x) q_i(t) \tag{9}$$

By applying boundary conditions ($u(0,t) = u(l,t) = \partial u(0,t)/\partial x = \partial^2 u(l,t)/\partial x^2 = 0$) and including just the dominant first three oscillation modes, the following equations are obtained for the vertical displacement $u(x,t)$, the slope $u'(x,t)$ and the velocity $\dot{u}(x,t)$ of any point x on the link at any time t:

$$u(x,t) = \varphi_1(x) q_1(0) \cos(\omega_1 t) + \varphi_2(x) q_2(0) \cos(\omega_2 t) + \varphi_3(x) q_3(0) \cos(\omega_3 t) \tag{10}$$

$$u'(x,t) = \frac{\partial u(x,t)}{\partial x} = \varphi_1'(x) q_1(0) \cos(\omega_1 t) + \varphi_2'(x) q_2(0) \cos(\omega_2 t) + \varphi_3'(x) q_3(0) \cos(\omega_3 t) \tag{11}$$

$$\dot{u}(x,t) = \frac{\partial u(x,t)}{\partial t} = \varphi_1(x) q_1(0) \omega_1 \sin(\omega_1 t) + \varphi_2(x) q_2(0) \omega_2 \sin(\omega_2 t) + \varphi_3(x) q_3(0) \omega_3 \sin(\omega_3 t) \tag{12}$$

subject to the initial conditions: $u(x,0) = \sum_{i=1}^{\infty} \varphi_i(x) q_i(0) = f(x)$; $\dot{u}(x,0) = \sum_{i=1}^{\infty} \varphi_i(x) \dot{q}_i(0) = g(x)$ (13)

The values of $q_i(0)$ and $\dot{q}_i(0)$ can be obtained from the normalised flexural stiffness as explained in [1].

The shear deformation force V acting in the link is given by V = EIρ. It has been shown [1] that the effect of this is to alter the frequency of the link oscillation modes and to reduce the slope at the end by an angle ψ given by:

$$\psi = V/KAG \tag{14}$$

where A is the cross-sectional area of the link and G is the shear modulus for the link material. K in (14) is a function of the link cross-sectional shape given by $K = AQ/I_a d$, where Q is the first moment about the neutral axis of the area contained between an edge of the cross-section of the link parallel to the main axis and the surface at which the shear stress is to be computed, I_a is the moment of inertia of the cross-sectional shape of the link computed with respect to its neutral axis and d is the width of the cross-sectional area at which the shear deformation is to be calculated.

3 Modelling of flexible joint

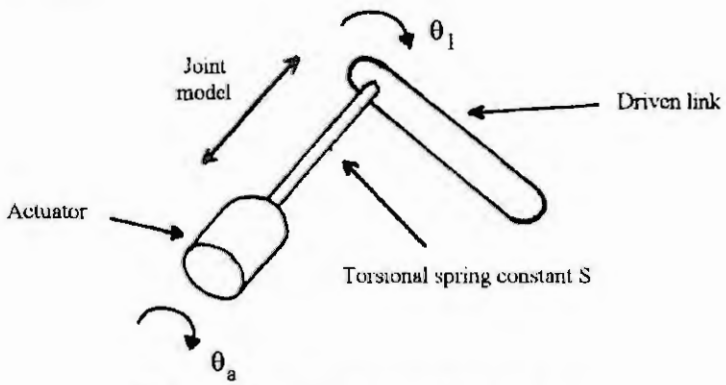


Figure 3: Representation of joint flexure

The basis of the flexible joint model used is that presented in [2], in which the joint is modelled as a torsional spring with spring constant S, as shown in figure 3. The kinetic energy in the joint can be written as:

$$K_j = 0.5\left(\dot{\theta}_a{}^T I \dot{\theta}_a\right)$$

The potential energy due to torsion in the spring is given by:

$$P = 0.5S(\theta_l - \theta_a)^T(\theta_l - \theta_a)$$

The Lagrangian for the system can then be written as:

$$L = K - P = 0.5\left[\dot{\theta}_a{}^T I \dot{\theta}_a\right] - S(\theta_l - \theta_a)^T(\theta_l - \theta_a)$$

The Lagrangian can then be solved to find the equations of motion. If T is the sum of gravity and other torques acting on the link, the static angular deflection due to torsion can be written as: $(\theta_l - \theta_a) = T/S$ (15)

The dynamic equation of motion of the joint can be written as: $I_{zz}\ddot{\theta}_l = S(\theta_l - \theta_a) + T$ (16)

4 Combined model for a two-flexible-link system

A basic structure for combining models of link and joint flexibility for a single-link system is described in [3]. The main difficulty in modelling multi-link flexible manipulators is that the rigid motion and the elastic motion are coupled together, and the elastic motion has direct effects on the transformation matrix between the link co-ordinates and the global co-ordinates. Due to the complexity of the problem, the modelling of flexible manipulators is initially simplified by neglecting the effect of the elastic motion on the transformation matrix and neglecting the effect of the elastic motion on the rigid motion. If the rigid motion is not affected by the elastic motion, the rigid system dynamic equations can be derived using the Lagrange-Euler principle. A further simplification can be achieved by considering only the first three flexible modes in the links, since higher modes have negligible influence on the behaviour of the system.

The equation of motion of a two-link system can then be written as: $u_j(x,t) = \sum_{i=1}^{3} \phi_{ij} q_{ij}(t)$, where j denotes the link number, i denotes the mode number, $u_j(x,t)$ is the vertical deflection of j at distance x and time t, and ϕ and q are shape and modal functions. q_{ij} can be obtained by solving the following differential equation:

$$\frac{d^2 q_{ij}(t)}{dt^2} + \frac{c_j}{m_{ij}} \frac{dq_{ij}(t)}{dt} + \omega_{ij}{}^2 q_{ij}(t) = \tau_j(t) \tag{17}$$

where c_j is the link damping coefficient, m_{ij} and ω_{ij} are the respectively the normalised mass and frequency of mode i for link j. The position vector is given by:

$$u_j(x,t) = \left[\phi_{1j}(x)\ 0\ \phi_{2j}(x)\ 0\ \phi_{3j}(x)\ 0\right] \times \left[q_{1j}\ \dot{q}_{1j}\ q_{2j}\ \dot{q}_{2j}\ q_{3j}\ \dot{q}_{3j}\right] \tag{18}$$

This can be used to give $u_1(x,t)$ and $u_2(x,t)$ as the dynamic deflection of links 1 and 2 respectively at time t.

For link 1, the total flexure-induced deflection at any point x along the link is given by $u_1(x,t) + u_{s1}$, where u_{s1} is the static deflection of link 1 due to its own mass and end-tip load plus the deflection due to the torsion in the joint. For link 2, the total deflection is given by $u_2(x,t) + u_{s2} + u_{s12}$, where u_{s2} is the static deflection of link 2 due to its own mass and end-tip load and u_{s12} is the deflection at the end of link 2 due to the flexure-induced slope at the end of link 1. The relevant parameters in these expressions are obtained from equations (10)-(12), (14)-(16) and (18).

5 Conclusions

The paper has explained the limitations of previous work modelling single flexible joint-link systems and has produced a model of improved accuracy. This has been incorporated into a two flexible joint/link system model, taking appropriate account of the dynamic coupling effects between the two links.

6 Future work

The presentation at the conference will show the performance of the model developed in response to various programmed trajectory demands. It is anticipated also that the current status of controller development for the two-flexible link system will be presented.

References

[1]. Morris, A S and Madani, A, 1996, 'Inclusion of shear deformation term to improve accuracy in flexible link robot modelling', Mechatronics, 6, pp 631-647.

[2]. Spong, M W, 1987, 'Modelling and control of elastic joint robots', Trans ASME, Journ of Dyn Sys, Meas and Control, 109, pp310-319.

[3]. Yang, G-B and Donath, N, 1988, 'Dynamic model of a one-link robot manipulator with both structural and joint flexibility', IEEE Conf on Rob and Autom, pp 476-481.

OBSTACLE AVOIDANCE FOR NON-POINT MOBILE ROBOTS

Bohumil Honzík[1] and Yskandar Hamam[2]
[1]Brno University of Technology
Božetěchova 2, 612 66 Brno, Czech Republic
http://www.fee.vutbr.cz/~honzikb
[2]Groupe ESIEE
B. P. 99, Noisy le Grand 93162 CEDEX, France

Abstract. The obstacle avoidance algorithm described in this paper is based on the generalized potential fields, which depend on robot's velocity as well as its position. Unlike classical potential fields, the robot is not repulsed by an obstacle if it is not moving toward it. The key problem when using generalized potentials is the computation of the time it would take to bring the robot moving with constant velocity to collision with the nearest obstacle in the given direction. It is easy for mobile robots with circular footprint, when orientation does not play a role. This paper presents new technique for computation of time remaining to the collision based on linear programming, which can be used in the case when robot's shape can not be approximated by a circle and rotational degree of freedom must be taken into account.

Introduction

Research in the area of obstacle avoidance may be divided into two classes : global and local. In global methods, geometric planning is performed ahead of path execution. The description of obstacles is assumed to be available to the path planner. The planner then produces a path from the initial state to the goal that avoids the obstacles. In local methods, sensory information about the local environment is used in real-time to generate a control input for the robot which brings the robot nearer the goal while avoiding nearby obstacles.

In the method of artificial potentials, the robot behaves like a charged particle that is attracted by the goal position and repelled by the obstacles. Krogh [1] introduced the important idea of a *generalized potential field* that is a function not only of the robot's position, but also of the velocity. With this concept, obstacles need not exert large forces on nearby points that are stationary or moving away from them.

For a mobile robot, the goal is to devise a strategy that will move the robot to its desired destination without colliding with obstacles. In addition, a robust obstacle avoidance scheme should be capable of dealing with moving obstacles. Using of generalized potentials results in shorter paths and leads to energy savings (mobile robots are mostly powered from the on-board batteries with limited capacity), more precise selflocalization (precision of dead-reckoning algorithms decreases as the length of the path increases) and time efficiency.

The following sections review the generalized potential field approach and show the difficulties encountered when applying this concept to non-point mobile robots. A solution to this problem using linear programming (LP) is then presented. Finally, grid-type map as proper world representation is chosen and its interconnection with the presented algorithm is proposed.

Generalized potentials

In contrast to classical potential fields, where the repulsive potential depends on the robot's position only and is given by the shortest distance to the obstacle, generalized obstacle potential is defined as the inverse of the *reserve avoidance time*, $\tau_M - \tau_m$, where the *maximum avoidance time* τ_M is the maximum time during which the velocity toward the obstacle may be brought to zero under constant deceleration without hitting the obstacle and the *minimum avoidance time* τ_m in which the velocity toward the obstacle can be brought to zero using maximum deceleration.

The key problem when using generalized potentials is the computation of τ_M. Most mobile robots used in research have circular, hexagonal or polygonal footprint that can be circumscribed and approximated by a circle. In this case the robot's orientation is not considered and solution of obstacle avoidance problem is simplified, because such robot can be modeled as a point in the plane, when the contour of obstacles is enlarged by the radius of the circumscribing circle. This method is called *configuration*

space approach and enlarged obstacles are *configuration space obstacles.* Computation of τ_M is then straightforward.

In some cases the robot's shape may not be approximated by a circle and rotational degree of freedom must be taken into account. In this case, the configuration space obstacles are of three dimensional forms which are not simple to represent or to calculate.

Solution using linear programming

In order to avoid computation of configuration space obstacles, calculations will be done in the *operational space* of task-related parameters (space with Cartesian rather than joint coordinates).

The task may then be stated as follows. Define the mobile robot as convex polygon and obstacles as constrained areas in the plane with coordinates x and y. Let the line perpendicular to this plane be the time axe t. We will call the new space *task-time space.* As the mobile robot moves in time, its edges form surfaces, which constrain three dimensional object. The same holds for the obstacles as well. If any of these objects intersect, collision occurs. Then, the intersection $\mathbf{p}_c = [x_c, y_c, t_c]$ with the lowest t-coordinate corresponds to the collision point, where x_c, y_c are its Cartesian coordinates and t_c denotes the time, when collision occurs. Situation is shown in Fig. 1a.

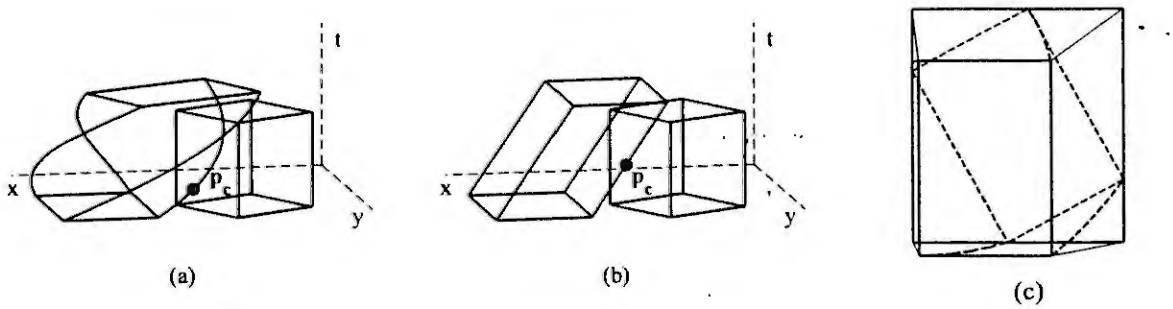


Figure 1: (a) Mobile robot moving in task-time space; (b) Robot translating with constant velocity; (c) Original task-time space obstacle (dashed) and its approximation (solid) - above view

Consider the simplified case, where all moving objects may only translate with constant velocity. Thus, the constraining surfaces are planes and linear programming can be used for computation of t_c. The linear program can be expressed as follows:

$$minimize \quad \mathbf{c}^{\mathrm{T}}\chi \tag{1}$$

$$subject\ to \quad \mathbf{A}\chi \geq \mathbf{b} \tag{2}$$

where $\chi = [x, y, t]^{\mathrm{T}}$ is the vector of variables to be solved for, $\mathbf{A}$ is a matrix of known coefficients and $\mathbf{c}$ and $\mathbf{b}$ are vectors of known coefficients. Inequality (2) is given by constraining planes which are formed by the mobile robot and obstacles (Fig. 1b). More specifically,

$$\mathbf{A} = \begin{bmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ \vdots & \vdots & \vdots \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} -d_1 \\ -d_2 \\ \vdots \end{bmatrix} \tag{3}$$

where

$$a_i x + b_i y + c_i t + d_i = 0 \tag{4}$$

is the equation of the i-th constraining plane in the task-time space. Because we are minimizing t, vector $\mathbf{c}$ in the objective function (1) is given as $\mathbf{c} = [0, 0, 1]^{\mathrm{T}}$. The number of rows in matrix $\mathbf{A}$ corresponds to the number of constraining planes and coefficients in (4) depend on robot's and obstacles' shape. In order to reduce the number of constraints in (2), we will convert the primal problem to its dual:

$$maximize \quad \mathbf{b}^T\lambda \tag{5}$$

$$subject\ to \quad \mathbf{A}^T\lambda \leq \mathbf{c} \tag{6}$$

Then the number of constraints in (6) is constant (three), while the number of dual variables λ varies depending on the number of constraining planes. The dual solution of the dual problem corresponds to the solution of the primal problem. This way we may calculate the time to collision, τ_C, when the robot is moving with constant velocity. It may be shown that $\tau_M = 2\tau_C$.

When considering rotational degree of freedom of the robot, situation becomes much more difficult, since objects formed by the robot moving with constant velocity in the task-time space are generally helicoidal-shaped and the problem is no longer linear. To avoid using of nonlinear programming we have properly approximated these complex objects. Instead of rotation of particular edges of the mobile robot, their movement is approximated by translation in the direction perpendicular to these edges, as shown in Fig. 1c.

The linear programming approach may be used for two convex objects only. If there are more obstacles, they must be treated separately. If either the mobile robot or obstacles are nonconvex areas, they must be decomposed in convex ones before. The described method may be used not only for one mobile robot moving among static obstacles but is suitable for the case of multiple moving objects as well.

Using evidence grids

In order to make the above described approach useful for the real system, proper world representation must be chosen. What we need now is the analytical description of robot's and obstacles' edges which must be gained from inaccurate sensor measurements.

Probably the best possibility is using of *robot evidence grids*. This approach allows to obtain reliable data from low cost and low precision sensors (e. g. ultrasonic and infrared). Another advantage is easy fusion of information from various types of sensors.

The evidence grid approach represents the robot's environment by two dimensional regular grid. In each cell is stored the evidence (or probability), based on accumulated sensor readings, that the particular patch of space is occupied. Given the robot's position, we increase the probabilities in the cells near the indicated object and decrease the probabilities between the sensor and the sensed object (since it is the first object in that direction). Detailed description of this approach may be found in [2].

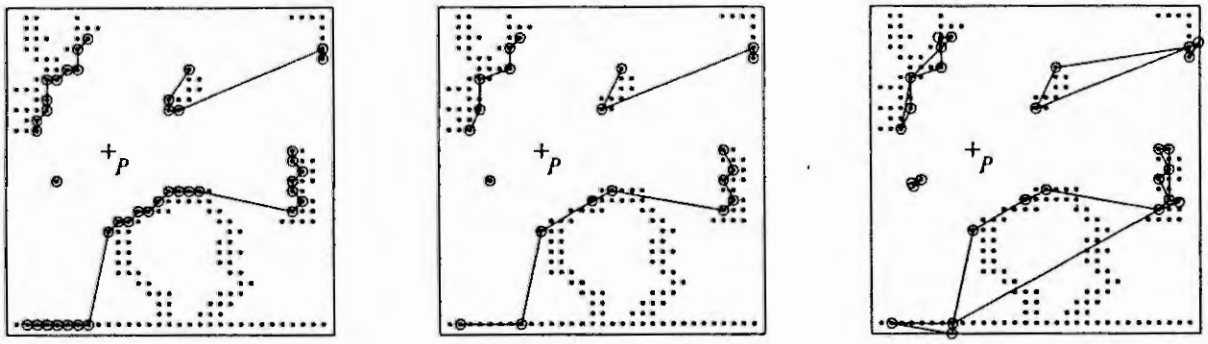
The evidence grid is a binary matrix. Elements assumed to be occupied correspond to ones, free cells are zeros. The task is now to gain from this world representation the analytical description of obstacles. All analytically defined (AD) obstacles must be convex areas. They need not to correspond exactly to the occupied grid cells, but should always contain them. If any occupied area in the evidence grid is nonconvex, it must be either decomposed in several AD obstacles or encapsulated in one AD obstacle.

The simplest brute force method is to treat every grid cell as one AD obstacle. Computing them is very easy, but there is one main drawback: as mentioned above, our approach to computing τ_M using LP can be used for two objects only. Thus, every occupied cell would have to be treated separately, which can take a lot of time. It is therefore desirable to collect as many neighboring cells as possible together and consider them as one AD obstacle. The following paragraphs briefly describe the procedure.

First, we need to compute the *constraining polyline* (CP), which is visible from the robot's point of view P (see Fig. 2a). We will *scan* the workspace by calculating all the cells lying on the line, which connects P with every cell of the workspace boundary. (This can be done using the slightly modified *Bresenham's algorithm*, which is well known from computer graphics.) The first occupied cell on this line corresponds to one of the vertices of the CP.

Then, some postprocessing follows. All possibly doubled vertices are removed. CP is decomposed into segments corresponding to particular obstacles. All vertices which are lying on the common line connecting two other vertices are also removed. If the grid resolution is too fine and therefore the number of vertices is too large, approximation of CP using Sklansky-Gonzalez' algorithm may also be done (described in [3]), see Fig. 2b. This technique measures the approximation error as the largest Euclidean distance between the approximation and the given digitized curve and removes all vertices which are in the given tolerance. (If set to zero, only vertices lying on common line are removed.)

After that, particular segments are decomposed into convex polygons (if necessary). Single points and lines are extended by two and one points, respectively, as shown in Fig. 2c.



(a) (b) (c)

Figure 2: (a) Calculating CP; (b) CP approximation; (c) Constructing AD obstacles

Applying to a mobile manipulator

The above described approach was applied to the control of the kineticallly redundant mobile manipulator (MM), which consist of a 6 DOF manipulator arm mounted on a 3 DOF mobile platform. The task considered in this study is that the hand of the MM is guided by a human operator while the platform is avoiding obstacles. Because the motion of the hand and interactions with the world are unknown a priori, the redundancy resolution has to be made locally (in real time) rather than globally. Rednndancy is resolved at the velocity level (see [4] for details):

$$\dot{\mathbf{q}} = \mathbf{J}_w^+ \dot{\mathbf{x}} + (\mathbf{I} - \mathbf{J}_w^+ \mathbf{J}) \left(-\frac{\partial \mathbf{h}(\mathbf{q})}{\partial \mathbf{q}} \right)^{\cdot} \tag{7}$$

where $\mathbf{q}$ and $\mathbf{x}$ are vectors of joint and task variables, respectively, $\mathbf{J}$ is manipulator Jacobian, $\mathbf{J}_w^+$ its weighted pseudoinversion (Jacobian is not sqnare matrix in this case), $\mathbf{I}$ is the identity matrix and $\mathbf{h}$ is the generalized potential. This eqnation is then integrated in real time. As the robot approaches the obstacle, the reserve avoidance time decreases and total force eventnally points away from the obstacle. When the platform tnrns away from the obstacle, the repnlsive force goes to zero (because there is no component of velocity in the direction of the obstacle) and the direction of the total force changes abruptly to point directly towards the goal. In order to avoid these oscillations, generalized potential $\mathbf{h}(q)$ in (7) is smoothed using low pass filter. Presented algorithm was simnlated using Matlab.

Conclusions

Collision avoidance of the omnidirectional mobile platform moving in a plane was solved. The described method is very fast, suitable for real-time applications and may be used with robot evidence grids. The presented technique may be extended to mnltiple moving objects and 3D world representation as well. New issues arise when considering mobile platforms with nonholonomic constraints. This is the subject of future work.

References

1. Krogh, B. H., A generalized potential field approach to obstacle avoidance control. In: SME Conference Proceedings, Bethlehem, Augnst 1984.

2. Martin, C. M., Moravec, H. P., Robot evidence grids. Tech. report CMU-RI-TR-96-06, Carnegie Mellon University, Pittsbnrgh, March 1996.

3. Sklansky, J. and Gonzalez, V., Fast polygonal approximation of digitized curves. In: Pattern recognition, Pergamon, 1980, Vol. 12, pp. 327 - 331.

4. Honzik, B. and Zezulka, F., Rednndancy resolution techniques for mobile manipnlators. In: Proc. 2nd International Symposinm Advanced manufactnring processes, systems and technologies, Bradford, UK, March 1999, pp. 79 - 89.

List of Authors

Contents

Abstracts of Posters

Abstracts of Papers

MODELING OF ION BOMBARDMENT OF SOLIDS USING MOLECULAR DYNAMICS

G. Betz *, C. Dandachi * and H.M. Urbassek**

* Institut für Allgemeine Physik, Technische Universität Wien, Wiedner Hauptstraße 8-10, A-1040 Austria

** Fachbereich Physik, Universität Kaiserslautern, Erwin-Schrödinger-Straße, D-67663 Kaiserslautern, Germany

Ion bombardment of a solid and the consequent emission of atoms from the solid (sputtering) is typically modeled by Molecular Dynamics(MD) simulations. The only physical input in such a model is the atomic interaction potential. Two independently developed MD codes (TB-KL and TB-W) are used to check the sensitivity of the results on the different computer-implemented numerics.

The codes TB-KL and TB-W are based on the same physical input (crystal size and geometry, ion impact points, interatomic potential, boundary conditions, etc.). Bombardment for 5 keV Ar $\rightarrow$ Cu(111) was compared in the present investigation.

The simulation crystallite consists of 6714 atoms arranged in 9 layers. 400 individual Ar ion impacts are investigated; the ions impinge in normal direction onto randomly selected impact points. Each simulation proceeds until 1 ps after the ion impact.

For the average quantities, like sputtering yield (number of emitted atoms from the solid per impinging ion), abundance distribution, energy distributions and yield of emitted clusters the agreement is very satisfactory between the two codes.

However, when considering individual trajectories, considerable discrepancies between the two codes could be detected. The correlation between the sputter yields, calculated for the same ion impact point, with these two codes is astonishingly poor. When calculating the linear correlation coefficient between these two data sets, it is only 0.61. We find that this disagreement is due to details of the integration routine used.

Using only the TB-W code, the correlation between results obtained when the time step in the integrator is halved is also quite poor (correlation coefficient 0.86). Note that in spite of the poor correlation, average results obtained by the two simulations agree very closely; as an example the sputter yields obtained are 14.03 and 14.16. Evidently such a detail in the integration routine as the exact choice of the time step can be responsible for a large part of the absent correlation between the TB-KL and TB-W results. We note that TB-KL used a second-order Verlet, while TB-W uses a fourth-order Gear integration algorithm.

It must be concluded that - while agreement between the *averages* over a large set of impact data is found - the correlation between the results of individual trajectories calculated with different codes may be not convincing.

The physical cause for this is the sort of chaos which is intrinsic to a classical mechanical many-body system in a multiple-collision situation. This disagreement may stem from details of the integration routine, which are hardly discussed even between closely cooperating research groups.

NEW PRODUCT DEVELOPMENT BY ARTIFICIAL AGENTS

T. Fent
University of Vienna
Brünnerstraße 72, A–1210 Wien

Neoclassic approaches assuming that there exists a stable equilibrium play a major role in today's business administration. Such approaches typically assume

1. perfect information about the analysed problem and its structure,
2. diminishing returns, and
3. only perfectly rational individuals.

However, in reality the individuals lack complete information and different participants interpret the same information in a different way. A dynamic market makes it difficult for the participants to maintain their competitiveness and survive in the long-run. A market built up by many individuals can be seen as a complex adaptive system (CAS). The main characteristics of CASs are:

open and dynamic: Only a closed system without external (exogenous) influences can tend to a stable-state equilibrium and persist there. In a CAS the individuals are always aware of unpredictable changes and adapt their behaviour whenever such shifts are encountered.

interacting agents: Decisions taken by one agent have an impact on the environment of all the agents and, in turn, they all have to adjust their strategy to the new situation. Thus, when one agent, due to evolution, changes her/his main strategy, this does not only effect the environment but also the evolution of the other agents. This causes the inherent dynamics of a CAS which makes it so unlikely to arrive to a steady-state. A typical example are the huge changes of stock prices. In a typical classical model they can only be explained by external perturbations, while in fact they are just a result of the investors' manners of trade.

emergence and self-organisation: There is no central planner deciding what and when to happen, but there are many intelligent individuals taking their own decisions. Only the whole bundle of actions and reactions can determine the behaviour of the system.

Practical examples of CASs are cities (not located in a dictatorial country), ecosystems, the internet, economies, and firms with a fractal structure.

Designing new products (and selling them) is often a process that involves several departments of a company. Moreover, the people dealing with all aspects of launching a new product have varying educational backgrounds and, therefore, do not speak the same technical language. In some organizations it can indeed be a big challenge to manage the communication required to capture all aspects of product development. However, nowadays decreasing product life-cycles require an efficient communication in order to launch new products before the competitors do it. An efficient information-flow is an indispensable prerequisite for a high speed of innovation.

The objective of the departments is to find decision-rules that yield low costs (also including opportunity costs). As a first approach the fitness of a rule is assumed to depend only on those costs that arise in that particular department. That means, all the departments try to find a strategy that minimizes their own costs without considering the indirect effects of their decision. However, in a more advances setup we also consider what happens if the fitness assigned to a certain rule is also influenced by the costs in subsequent departments.

In the simulation we tried 432 different parameter settings. It turned out that that modelling the rule-base as a classifier system and using a genetic algorithm to train the rule base indeed led to remarkable reductions of development costs. Certainly, it would be possible to implement a CS which always finds the optimal solution by increasing the population size until one rule for each possible input is available. However, the purpose of a CS is finding a generalization among the encountered inputs, and establishing a simple policy with satisfying results. Obviously this also conincides with a real-world situation where even the most intelligent individuals only use a limited number of thumb rules to take their decisions.

Genetic Portfolio Optimization in Models for Financial Markets

Günther Grohall, Siegfried Wassertheurer
Technical University Vienna, Dept. Simulation Techniques
Wiedner Hauptstraße 8-10, A-1040 Wien

Abstract

The aims of this short paper are to show the reader the importance of portfolio management in financial applications, the difficulties in assembling them and our way for a solution.

Introduction

In financial engineering, the problem of creating a portfolio is a very challanging one. The thing about it is to pick a few stocks out of a "pool" of financial titles which work together in an especially harmonic way. The two properties, which are looked at in this work, are mean and variance of the built portfolio. It is rather easy to get the mean; it is just the weighted average of all the means of the stocks within the portfolio. To calculate the variance is slightly trickier due to mutual impacts of the stocks, which makes it possible that the random changes of the courses are at least partly extinguished. Because of this effect, the risk of the portfolio sinks below the weighted average risk of the single stocks!

Problem Description

Since all stocks have different impacts on each other, one has to try out every possible combination to find the best. The effort of calculation rises polynomically with the number of stocks in the pool and the portfolio.Due this fact one reaches the maximum capacity of today's computers' calculation power immediately, even small samples selected. In principle, an algorithm having smaller expenses is to be preferred.

At this point Genetic Algorithms enter the scene. This method, invented some 25 years ago, tries to apply natural evolution's operations on the problem to be solved. It can be used universally and has its best performance when used on discrete problems. Finding an optimal portfolio is such a discrete problem, since certain stocks have to be taken out of a finite number of stocks.

Within a short amount of time even a huge solutionspace can be searched efficiently. Applied on certain problems, they can find the optimal solution long before a stochastic or analytic method will do. Applied on large problems, results near the global optimum will be found soon.

Conclusion

The issue of this work is not a big surprise but anyway very fine to be seen. Compared to the pure stochastically working Monte-Carlo-Method Genetic Algorithms deliver significantly better solutions. Tests on several problems showed that Genetic Algorithm's perform astonishingly high.

References

[1] Jochen Heistermann, *Genetische Alogrithmen - Theorie und Praxis evolutionärer Optimierungen* , B.G. Teubner Verlagsgesellschaft, 1994

[2] Jack Clark Francis, *Invesments, Analysis and Management*, MacGraw-Hill, 1991

Kernel Methods for Generation of Pseudorandom Numbers in Models of Discrete Processes

M. Klug
Department for Simulation Techniques, Vienna University of Technology
Wiedner Hauptstraße 8-10, A – 1040 Vienna, Austria
mklug@osiris.tuwien.ac.at

In Discrete-Event Simulation a model is based on a given sample of individual identically distributed data from an observation. Basing on the central limit theorem of statistics, these values of the sample have been pre-processed and implemented into the simulation model by calculating mean and variance of the sample and using the implemented distribution function. Distributed and online simulation increases the demand of a new, preferably nonparametric computation method for those parameters.

Investigating Kernel Methods for density estimation, an estimated density function can be calculated directly from the sample $x_j \quad j = (1,...N)$ using a Kernel Function K and a smoothing factor h through:

$$f(x) = \frac{1}{Nh} \sum_{i=1}^{N} \frac{K(x - x_i)}{h}.$$

The formula above might be used, if for all $x \in [x_{\min} - h, x_{\max} + h]$, $f(x)$ will be calculated and a new random number is computed by using table lookup method. Depending on the size of the sample and the length of the interval, this will take a long time, for pre-processing and is not useful, if the values for the given sample x_j are dynamically changing.

Therefore two easy to use methods are developed in the statistics for calculating a new random number y, the Order Statistics Method and the Rejection Method.

1. Compute a uniform distributed discrete random number $z \in \{1,...,N\}$
2. Compute a new random variable w by
   2a. Rejection Method:
   - Compute a uniform random variable $u \in [0,1]$ and an independent uniform distributed random variable $w \in [-1,1]$.
   - If $u \le 1 - w^2$, take w, else compute a new pair (u, w).

   2b. Order Statistics Method:
   - Compute three independent uniform random variables $w_1, w_2, w_3 \in [-1,1]$.
   - If $|w_3| \ge |w_1|$ and $|w_3| \ge |w_2|$ then $w = w_2$, else $w = w_3$.
3. The new random number y is given by: $y = x_z + hw$

Both methods are implemented in SLX and original samples are calculated with different size (N = 20, 50, 100, 500). Those samples are taken to calculate new samples with various different smoothing factors h. Taking each new sample, the mean and the variance is calculated again, and the differences between mean and variance are stored. Mean and variance of the differences are calculated again for both methods.

It can be shown that the given sample might be taken as basis for the new random numbers, if the smoothing factor is specified correctly.

A problem arises if the density function is limited at least at one end (like the Poisson Distribution, or the Triangular Distribution for both ends). Then the kernel methods still calculate also numbers outside the valid interval, which doesn't arise using the original density function. It is not a matter of choosing the right value for h, instead. Additional checks have to be implemented to avoid such problems. This point has to be investigated further on.

Summarising the results, Kernel Methods can be used as an efficient alternative for computing pseudorandom variables. Implementing them the mentioned problems (choosing the smoothing parameter and exceeding the limitations) might arise and have to be kept an eye on.

References:

Devroye, L.; Gyorfi, L.: "Nonparametric Density Estimation, The L_1 View" 1985, John Wiley & Sons
Silverman, Bernard W.: "Density estimation for statistics and data analysis" 1996, Chapman and Hall

"Improved elastic rod model of 3D RNA structure"

E.E.Kozyreva	State Research Institute of Genetics and Selection of Industrial Microorganisms, 113545, Russia, Moscow, 1st. Dorozhny proezd., 1
E.I.Kugushev	Institute of Applied Mathematics by M.V.Keldysh of Russian Academy of Science, 125047, Russia, Moscow, Miusskaya pl., 4
A.V.Maikov	Institute of Applied Mathematics by M.V.Keldysh of Russian Academy of Science, 125047, Russia, Moscow, Miusskaya pl., 4
E.L.Starostin	Institute of Applied Mathematics by M.V.Keldysh of Russian Academy of Science, 125047, Russia, Moscow, Miusskaya pl., 4

ABSTRACT

A problem of reconstruction of approximate large-scale 3D structure of an RNA molecule from its secondary structure and latest experimental and theoretical data on thermodynamic stability of its structural elements is considered. Both a new mathematical model and its computer implementation are presented. An RNA molecule is treated as a set of basic structural elements (dangling ends, stems and loops of various types) modeled by elastic rods with different elastic parameters depending on the primary structure of the corresponding structural element. A numerical procedure is developed for computation of shapes of the RNA elements and for assembling the whole molecule. The comparison of the available X-ray diffraction analysis results (yeast phenylalanine tRNA) with the proposed rod model reveals a good correspondence of the overall tracing of the polynucleotide chain.

IMPLICIT SOLUTION OF KINETICS IN HV CIRCUT BREAKERS

J. Kunovský, P. Pospíšil and P. Sezemský
Department of Computer Science
Faculty of Electrical Engineering and Computer Science
Brno University of Technology
Božetěchova 2, 612 66 Brno, Czech Republic

Introduction

Improving the function of high-voltage (HV) circuit breakers filled with SF_6 consists of modelling phenomena that come into play during arc quenching after current zero. One of these phenomena is the kinetics of chemical reactions in a system S of dissociation and ionization products of the quenching medium during a sudden drops in arc temperature from the plasma state to the gas state at low temperature.

Kinetic reactions in S are mathematically modelled by a stiff system of nonlinear ordinary differential equations.

Implicit Modern Taylor Series Method

A new modification of the Modern Taylor Series Method - Implicit Modern Taylor Series Method has been developed especially for stiff systems (Figure 1).

Using the forward method (from the previous samples) the derivatives up to the n-th order at the point A are calculated. Knowing the higher derivatives, it is now easy to perform the calculation with an integration step H.

The obtained value of of the solution (point B) is used as the starting point of a new calculation. Using the backward method (from the subsequent samples) the derivatives at point B are calculated. Knowing the higher order derivatives, a calculation is performed with the integration step (-H).

If the obtained value (at point C) has a larger computation error than the required accuracy ϵ , a new calculation is started from point A with the integration step H/2.

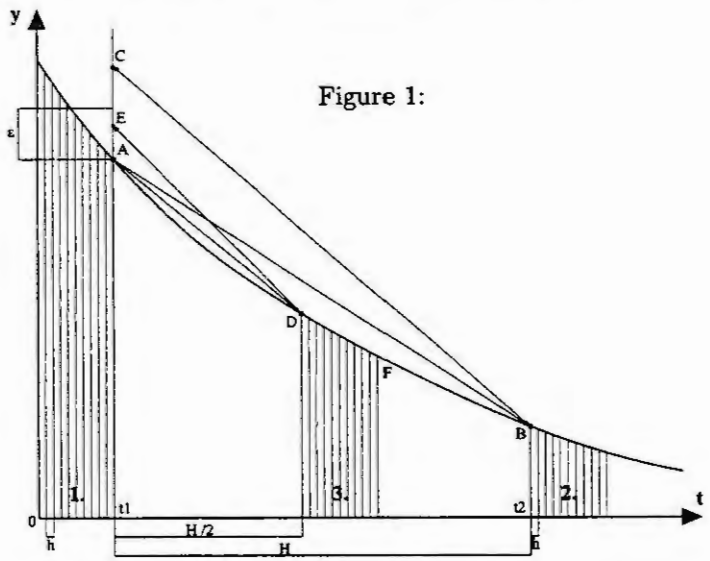


Figure 1:

Conclusions

A detailed description of the problem will be presented during the conference.

References

[1] Kunovský, J.: Modern Taylor Series Method, Habilitation work, TU Brno, 1995

The project was supported by the GA 102/99/1499 and research program CEZ: J22/98: 262200010

Approach to Combined Modelling and Simulation at Algorithmic Level

M. Lingl, EVVA-Werk, Vienna

As today's computers have by far enough calculation power to simulate large models with different levels of modelling packed together, combined models with continuous and discrete elements are no longer special cases but rather normal. But most simulation programs were initially design to solve only one kind of model. Features for handling different models were added later with some difficulty, which made their use rather complicated. This distinction was supported by the fact that most simulationists were experts for either kind of models and regarded the "other side's" work with suspicion.

By observing both kinds of models at the algorithmic level, at least some technical problems can be solved. Maybe a new view of the problem can even overcome some biases.

The Simulation Step

The concept that both kinds of simulation have in common is the *simulation step*. There is no other way to proceed by means of a digital computer.

Discrete models rely heavily on this concept, although the corresponding object is normally called *event* there. Events are written into an event list, they are sorted there by the simulation time they are to take place (sorting algorithms may only differ in the way they treat concurrent events) and finally they are executed. New events may be added at any time, and events that were not yet executed may be deleted. It is easy to see that the same method works if the word *event* is simply replaced by *simulation step*.

Continuous modelling and simulation is based on the numerical solution of differential equations. Usually, algorithms are used which move forward in time, carefully stepping from one integration step to the next one, in order to generate (discrete) data points as close to the continuous function as possible, thereby maintaining the impression of a smooth function. Events in continuous simulation are normally considered as something weird, which disturbs the (imaginary) "flow" of the integration algorithm.

The integration algorithm drives the simulation, which not only makes it very difficult to implement discrete elements, but also to combine two different integration algorithms to one model. It is possible to split the algorithm into two parts, one determining the step sizes, the other one performing the summation, so that the first part schedules the steps into a list of simulation steps, which is worked out by the second part.

This strategy makes it possible to let different kinds of simulation algorithms, both continuous and discrete, write into the same list of simulation steps. As any simulation step, independent of its origin, is allowed to alter or remove any of the other scheduled simulation steps, any range of interaction between the different submodels, as well as any kind of hierarchy among them, is feasible. As a side effect, submodels being combined that way are automatically synchronised, as their respective events are always executed in the correct order.

Although a simulation program fully supporting this concept (an object-oriented approach seems to be fit) is yet to be implemented, only little effort is needed to transform existing models in a way that event lists following these guidelines are generated in a simulation system. This method was applied successfully when creating a model of a biological waste treatment plant at the Austrian Research Center Seibersdorf.

THE ELECTRICAL STIMULATION OF THE SPINAL CORD: COMPUTER MODEL FOR ELECTRODE POSITIONING

Karen Minassian, Frank Rattay and Harald Markum

TU-BioMed, Vienna University of Technology, Austria

The electric field as applied in epidural spinal cord stimulation for paralyzed people is calculated with the Finite Element Method using ANSYS software. In a second step the excitation of selected target neurons is computed with ACSL by evaluating a compartment model for myelinated nerve fibers. The simulations demonstrate that the relative location of implanted electrodes with respect to the segments of the spinal cord can be identified.

Clinical application of the electrical stimulation of the spinal cord with implanted electrodes cause a great variability in muscle responses including single twitches, constant tonus, suppression of spasms or generation of rhythmic activities. The stimulating electric fields for these applications are usually generated by implanted multichannel electrodes that can operate in different modes.

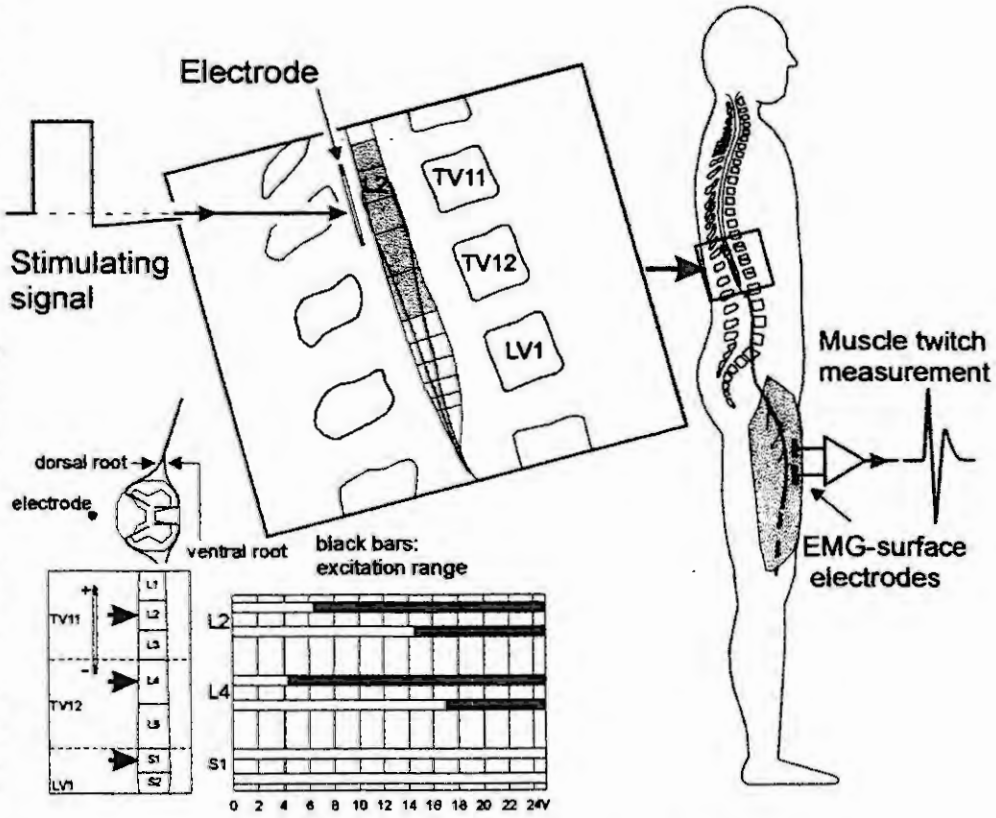
In this study we develop a model that investigates the influences of electrode position and different stimulating monophasic and biphasic pulses on the excitation process and on the order of the excitation of the target neurons.

The estimation of thresholds for a selected ensemble of target neurons is done in a two step procedure: The first step is to compute the electric field generated by the electrodes, using the Finite Element Method. In the second step the calculated voltage profile along a target neuron is used as input data for an electric network model that represents the spatial form of the neuron including the nonlinear membrane conductance for ion currents.

The Finite Element geometry of the spinal cord is modeled by creating volumes from several cross section pictures of the 'Visible Human Male'. The electrical excitability was evaluated for target neurons representing ventral and dorsal root fibers of the L2, L4 and S1 segments.

The upper figure explains the electrical stimulation of nerve fibers arising from the lumbar portion of the spinal cord (L2) and the corresponding muscle response.

The lower figure shows activation of spinal root fibers for an electrode at L2 spinal level with 210µs pulses. Arrows mark the entry levels into the spinal cord of the 6 simulated target fibers with the threshold values given in the corresponding right pictures. The upper and the lower black bar of each segment mark the computed excitation range of the dorsal and ventral roots, respectively.



The simulations demonstrate a strong relation between electrode position and the order of muscle twitch responses. Thus, we support the surgeon with an additional tool to identify the relative location of implanted electrodes with respect to the segments of the spinal cord. For lumbosacral spinal cord stimulation we found the following rule: Dorsal root fibers have the lowest threshold values, ventral root fibers are more difficult to excite and dorsal columns are not excitable within the clinical range of 10 Volts.

An extension of Grodin´s model for the pulmonary system

N. Popper, A. Pelikan (Wien, A), J. Krocza (seibersdorf, A), B. Bracio (Univ. of Idaho, USA)

Part of PhySim
Simulation of physiological events

Within the project PhySim, a cooperation between the Austrian research center Seibersdorf - department for medicine and rehabilitation technology - and the technical university of Vienna - working group simulation News - the physiological events of the human body are modeled and simulated in different programming languages.

Introduction

Physiology examines the interactions between cells, tissue, organs, organism and the environment. Life of every organism is determined by various processes like diet, growth, development, propagation and dying. Basics for these events are the synchronisation of energy or substance transport, signal transfer and control. Because of these reasons the organism is possible to maintain inner, dynamic balance. This condition described as homöostasis depends on various environmental influences which have effects on the organism, and therefore have to be considerated in the modeling process.

Respiration

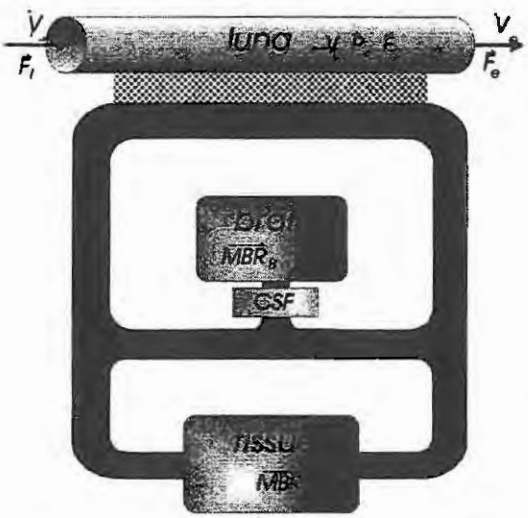


fig. 1: structure of the resp. system

The first dynamical model of the human respiration was described in an article by Grodins (1954). This model was improved over the years, using the newest knowledge of simulation and medicine.

Main Issue of the respiratory system is the transport of oxygen. Corresponding to this function the model consists of 4 parts

- convection within the lung
- diffusion from alveols to blood
- convection from lung to tissue
- diffusion from blood to sells

The structure of the respiratory system,, with two compartments, brain and tissue, is shown on fig. 1. The 2nd part of the model is the controller, using different parameters to create a specific pattern of breathing.

Extensions

Using MATLAB, the newest implementation substitutes the original continous breathing flow by a realitstic pulsatile flow. The implementation using MATLAB Simulink is shown on figure 2. Every part of the model shown on figure 2 represents a part of the windpipe. Another extension is to divide the lung in different parts, following the phsiological structure of the lumbs. This segmentation is usefull because of the influence of the perfusion within the lung.

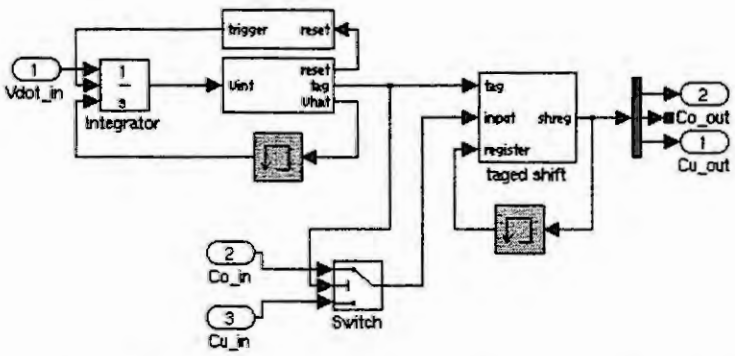


fig. 2: SIMULINK Block representing a part of the windpipe

Modelling of Dynamic Systems by means of Relational Databases

Thomas Preiß, Felix Breitenecker
Technical University Vienna, Dept. Simulation Techniques
Wiedner Hauptstraße 8-10, A-1040 Wien

Abstract

Within this short paper we introduce a concept, which embeds dynamic modelling in a tuple calculus. We combine the theory of relational databases with our calculus to get mechanisms for the joining of models. Also a formal language to use this tuple calculus is described.

1 Introduction

In data processing sophisticated systems are developped, which store and manage large amounts of data. DataBase Management Systems (DBMS) are able to create relations between special kinds of sets, and can access data directly. A model of continuous simulation based on the theory of sets is presented. The different object types are formed to tuples and stored in the tables of the DBMS.

2 The concept of the *Data Based Model Relation*

We define a kind of hierarchy of the elements of the tuples. The basic elements are called elements of *atomic sets*. The tuples of these elements are elements of *state sets*. Within a cartesian product these sets form the *state space*. We say, that models are unique functions of a tuple of elements of *atomic sets*, which is called the *model parameter set*, to elements of the *state space*.
Two other functions are also part of the concept. The *equation selection function* φ_m makes a selection of model equations dependent on the assignment operations of the model parameter set, which are represented by the domain of φ_m. A selection by parameters is also possible with the *set selection function* $\bar{\sigma}$.

2.1 The Model Query Language - *MQL*

A formal language is defined to calculate the model building operations, which consists of joining models, building models, exporting models and calculating simulation results. The algorithms are based on the tuple calculus joined with operations done by relational databases.

2.2 The calculating mechanism *data based integration/data based calculation*

Due to the usage of a DBMS the results of a model simulation are stored in the tables. It is now possible to examine the structure of a simulation problem and then to look for results of similar problems. These results can be interpolated to achieve solutions for our problem.

3 Conclusion

Relational database management systems are widly used and available. With the concepts of *DBMR* and *DBI/DBC* a method is shown, how to use RDBMS as a fundamental part of continous modelling and simulation.

References

[1] F. Breitenecker: *Models, Methods and Experiments – a new structure for simulation systems.* Mathematics and Computers in Simulation Bd. 34, 231-260.

[2] F. Cellier: *Continuous System Modeling.* Springer 1991.

[3] E.F. Codd: *A Relational Model of Data for Large Shared Data Banks.* Communications of the ACM 13, 377-387, 1970.

An SLX - Toolbox for Enhanced Coloured Petri Net Modelling

S. Rahmi, M.Klug
Department for Simulation Techniques, Vienna University of Technology
Wiedner Hauptstraße 8-10, A – 1040 Vienna, Austria
shaby@osiris.tuwien.ac.at; mklug@osiris.tuwien.ac.at

The goal of this project was to create a toolbox for working with Petri Nets in SLX. SLX itself is a new textual based discrete event simulation environment. Because of its flexibility and extensibility the library could be implemented with a calculable amount of effort and time. Developing the toolbox was an additional test of possible shortcomings of this state of the art simulation tool.

Commands are defined in statement definitions, where each of them calls a procedure handing over the parameters. The implementation of the procedure directly as a statement raised unsolvable problems, caused by it's complexity, so the commands themselves are defined in statement definitions, their functionality in the procedures called there.
The commands "`create_transition`" and "`create_place`" with their individual required parameters generate places and transitions. These may be connected via an arc using the function "`create_arc`". Additionally, not arcs are also implemented. All events taking place in a Petri Net are implemented in procedures, i.e. "`mark_place`" offers the possibility to place tokens into a named place.

Using the toolbox for Petri Nets turns out to be working in two steps: First the user creates a Petri Net using the commands the toolbox offers . This information is stored and the Incident Matrix is calculated in a second step, where the whole model runs without affecting the user.

One main problem during the implementation was the need of an internal structure, where all places, transitions and arcs get their own unique identity that can be used in all functions further on, without the user having to worry about it. That was solved on two levels with arrays. All places, all transitions and all arcs are written into arrays, therefore each element is assigned a lower level index to be called on. That means i.e. a place called *junction1* by the user is internally called *place[4]*.

The Incident Matrix of the Petri Net is calculated in two steps: Each transition is checked for a connection to each place. If there exists an arc pointing from transition to place the weight of this arc is written into the corresponding place in a matrix called "`cminus`", else 0. In the case of enhanced Petri Nets, non arcs are also considered. In step two each place is checked for connections to each transition, the result is written into a matrix called "`cplus`". Both matrixes are stored for further use during the run.

"`Fire_transition`" offers the possibility to fire one chosen transition, "`fire`" starts a repeated discrete distributed firing. After each time a transition is fired the current marking is written into a vector with size = number of places. These changes are easily determined by using *cplus* and *cminus*: subtracting *cplus[i]*, where *i* is the index of the fired transition, from the *marking_vector* and adding *cminus[i]* to it results to the updated *marking_vector*. This vector is generally used to check if any transitions are enabled or not.

Of great interest is the recognising and treatment of Deadlocks. Deadlock means the system is in a condition where not any event can take place anymore, concerning Petri Nets it means, that no transitions are enabled. For this case a function exists that checks each transition of being enabled or not. Into the corresponding places of a vector of size = number of transitions, one or zeros are written. If the norm of the vector is equal to zero the system has reached the state of deadlock.

This check is made by default every time a transition is fired, but the user is free to use it himself.

If a deadlock is detected the execution of the program is stopped and an error is reported.

Another check that is done every time a transition is fired is for repeated marking. If the marking of the system has occurred once before a message is put out, telling the user so. It is for the user to decide to continue the simulation or to stop it.

After the library for Petri Nets was completed, one started to extend the functionality to enhanced coloured Petri Nets. Basically adding colours means to extend each element with one additional dimension.

A problem that occurred is that SLX does not support dynamical arrays in the way C does. Arrays must have a fixed dimension at the time of initialisation. That brings about that either the user has to change the dimensions in the library as needed, or very big dimensions are given beforehand what causes a lot of wasted memory and the creation of huge matrixes causing to slow down the speed of calculation.

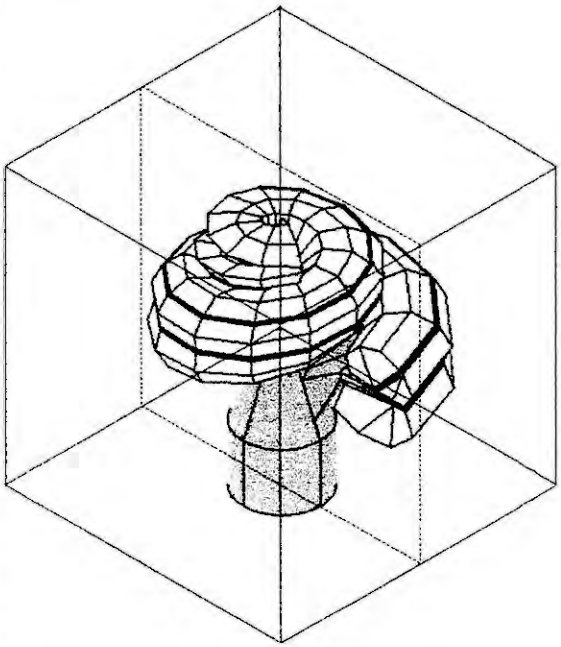
THE ELECTRICALLY STIMULATED HUMAN COCHLEA: CALCULATION OF THE POTENTIAL DISTRIBUTION IN THE INNER EAR AND THE EXCITATION OF THE AUDITORY NERVE

F. Rattay*, P. Lutter* and R. Naves Leao**

*Vienna University of Technology, Austria
**Medical School Uberlandia, Brazil

frank.rattay@tuwien.ac.at

The potential distribution in a simplified spiraled model of the electrically stimulated human cochlea is calculated with the finite element technique. In a second step the artificially generated spiking activities in selected target neurons are simulated with compartment models. Computer simulations demonstrate that the rather long peripheral process of the cochlear neuron (which is typical in man but not in animals) causes first excitation in the peripheral part of the human cochlear neuron; excitation of the central part needs higher stimuli. This result is just the opposite of single fiber measurements in animals.



In patients with complete hearing loss auditory sensations can be restored by electrical stimulation of the cochlear nerve. Stimulation via multi-channel electrodes inserted into the cochlea (auditory part of the inner ear) leads to a quite different nerve spiking pattern compared to the natural case. The development of new speech processing strategies calls for a tool to predict the spiking pattern in the electrically stimulated human cochlea.

In animal experiments measurements with microelectrodes inserted into single fibers of the auditory nerve show the relation between stimulation parameters and the generated spiking pattern. Our study demonstrates the importance of modeling work and computer simulations because such experiments cannot be done in cochlear implant patients – and the unique human cochlear morphometry does not allow to generalize the available animal data.

We reconstructed the spiraled cochlear shape from a picture of the central cross section. The gray tone intensity in the figure is proportional to the resistivities of the different cochlear regions. In the first step the electric field was determined by a finite element program. In the second step the voltage profiles along selected target neurons were used as input data for a compartment model of the human cochlear neuron. All subunits of a single neuron were individually described by connected electric circuits. The ion channel dynamics in the neuron's membrane were simulated with a modified Hodgkin Huxley model: Tenfold ion channel density was assumed in the nodes of Ranvier and the gating mechanism was speeded up by a factor k=12. For additional information see [1,2]. The electric field was evaluated for 18 target neurons, each of them separated by a turn of 30 degrees along the cochlea.

The simulations demonstrate that in man the peripheral parts of neurons close to the electrode are most excitable. Increase of stimulus intensity causes a second point of spike generation in the central part of the neuron. Another essential difference in the human spiking pattern is caused by the unmyelinated region around the soma that is typical for man only. A peripherally initiated spike arrives in the central nervous system with a time delay of about 400 us compared to a spike already generated in the central process, which is in good agreement with data from human screening techniques. The explanation for this phenomenon is that spikes already generated at the central side do not have to pass the time consuming way across the unmyelinated soma region. The human cochlear implant user will obtain a temporal pattern far different from what is seen in animal experiments. Consequently, it is questionable to apply data about the temporal fine structure in the spiking pattern measured in animals to human patients.

The first successful combination of the simulated human potential distribution in the cochlea with a nerve model including recent morphometric human data demonstrates essential differences between the spiking behavior in humans and animals. The proposed method is of use for the prediction of electrically generated spiking patterns for arbitrary electrode configurations.

[1] F. Rattay. 1999. The basic mechanism for the electrical stimulation of the nervous system. Neuroscience 89, 335-346

[2] F. Rattay, R. Naves Leao and P. Lutter 1999. The electrically stimulated human cochlea: a simulation study of the potential distribution and excitation of the spiral ganglion. Med. & Biol. Eng. & Comp. 37, Suppl. 2, 802-803

MECHANICAL MODEL FOR A FLAT STRUCTURAL INHOMOGENEOUS AND ANISOTROPIC TISSUE

Reihsner, R., Beer, R.J., Gingerl, M. & Millesi, H.

1. INTRODUCTION

Real tissues are usually composed from different materials, pieces of threats randomly distributed in a homogeneous matrix. The distribution of the threats may lead to a isotropic or generally anisotropic behaviour of the tissue. Generally, the distribution of the threats and the properties of them are unknown. A determination of this distribution by, for instance optical, inspection is usually often for some reason not possible. Even in the case that the microstructure is known, we need for a continous-mechanical treatment the develpoment of a continous model. Instead of such an inspection, in this paper a continous model is determined directly from the deformation analysis applying a radial symmetrical deformation to the flat specimen.

2. EXPERIMENTAL PROCEDURE

Figure 1 shows a scetch of the measurement procedure. The forces necessary to apply a radial symmetric deformation are measured in six directions by special load cells (Reihsner et al, 1998).

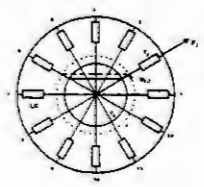


Fig. 1 Scetch of the loading device The situation is shown for the main direction 2 (LC: load cell).

Fig. 2. Biaxial testing equipment fo soft tissues.

For the case of a viscoelastic behaviour of the specimen a relaxation of the applied forces occurs. This relaxation can be measured until it is practically over (that means the relaxation phase is observed until changes of forces cannot be resolved anymore. The time necessary for this process is of course different for different specimens.). Only the initial and the final values of these forces are mentioned in this paper. The final values are regarded as the elastic response of the specimen (Dunn & Silver, 1983, Reihsner & Beer, 1995). The forces F_i measured in any directions i (i,j=1,..,6) is the resultant of all forces f_j, acting in the different directions (1).

$$F_i = \sum_1^6 f_i \cdot \cos(\alpha_{ij}) \qquad (1)$$

represents a set of 6 equations for 6 unknown values f_j.

Figure 2 shows a photo of the biaxial testing equipment (Reihsner et al, 1998). The equipment (Fig. 2) consists of 12 halfaxes, in each of which is a load cell integrated. The process is displacement-controlled by computer.Figure 3 shows a typical measurement F_i ($\varepsilon=\Delta r/r_o$). Δr denotes the increase of the radius; r_o denotes the reference value of the radius. The $F_i - \varepsilon$ relationship shows a nonlinear starting region which indicates that at the begining not all threats (fibers) are loaded. With increasing strain the number of loaded fibers is increasing, too leading to a more or less linear response to further deformations. For each graph (i=1..6) a value ε_i^* can be determined to devide the nonlinear and the linear region. For strains $\varepsilon \geq \max(\varepsilon_i^*)$ we may assume that all threats (fibers) are participating on the process.

3. INTERPRETATION OF THE MEASUREMENT

The interpretation of the measurement is strongly influenced by the structure of the specimen itself. In case of living tissues we may assume that the threats (fibers) are embedded in a high viscous ground substance matrix. That means that we may in this case assume that the forces acting in the single fibers are not influenced by those of the other fibers and they may move through the ground substance independently from one another. Under this assumption, the values of f_j represents the stiffness distribution in the specimen under consideration. With the additional assumption that the mechanical properties (Young's modulus) of the different fibers are more or less equal this result represents the network of fibers by the density ν and the cross-sectional area of a singel fiber A_{Fiber} of a theortical network of straightened fibers in different directions:

$$\sigma(\alpha) = F(\alpha) / (A_{Fiber} \cdot \nu(\alpha)) \qquad (2)$$

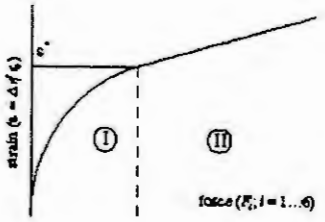


Fig. 3: Typical graph of the F_i - ε relationship. Region I. increasing number of strained fibers; region II: all fibers are strained.

4. APPLICATION EXAMPLE

Figures 4.1 and 4.2 show the result of the modelling in the described way for two specimens of human skin taken from the abdominal region and the forearm (palmar, 10 cm distal from ellbow), respectively. At ε^*=12.5% the linear region (Fig. 3) was reached in both cases.

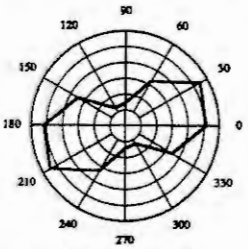


Fig. 4.1. density distribution $\lambda(\varphi)$ of the fibers as a function of the direction in skin from the abdominal region for the radial strain level r/r_0=1.10.

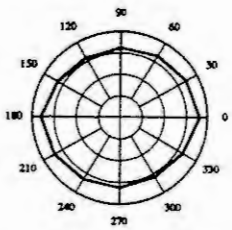


Fig.4.2. density distribution $\lambda(\varphi)$ of fibers as a function of the direction in skin from the forearm or the radial strain level r/r_0=1.10.

A comparison between these two results shows that in case of skin from the abdominal region (Fig. 4.1.) we have a fiber distribution with a preferred orientation whereas in case of the specimen from the forearm (Fig. 4.2.) we have a more or less uniform distribution of fiber directions. In terms of Young's moduli skin from the abdominal region shows a pronounced orthotropic behaviour whereas skin from the forearm display nearly isotropic behaviour.

5. REFERENCES

Dunn, M.G. & Silver, F.H. (1983). Viscoelastic properties of human connective tissues: Relative contributions of viscous and elastic components. *Connective Tissue Research* 12., (January 1983) 59-70.
Lanir, Y. (1979). A structural theory of the homogeneous biaxial stress-strain relationship in flat collagenous tissue. *Journal of Biomechanics* 4., (August 1979) 423-436.
Reihsner, R. & Beer, R. (1995). Elastizität menschlicher Haut bei zweidimensionaler Beanspruchung. . *Österreichische Ingeniuer- und Architektenzeitschrift* 140., (September 1995) 299-303.

Reihsner, R., Beer, R., Eberhardsteiner, J. & Millesi, H. (1998). Neuentwicklung einer Prüfmaschine für zweiachsige Festigkeitsuntersuchungen an anisotropen weichen biologischen Geweben mit Anwendungsbeispielen. *Österreichische Ingeniuer- und Architektenzeitschrift* 143., (June 1998) 262-266.
Authors: Dr. Roland Reihsner, Ludwig Boltzmann Institut für Experimentelle Plastische Chirurgie
Prof. Hr. Dr. Rudolf Beer, Institut für Festigkeitslehre der Technischen Universität Wien, A-1030 Wien, Adolf Blamauergasse 1-3
Dr. Manfred Gingerl, Institut für Festigkeitslehre der Technischen Universität Wien
em. O. Univ. Prof. Hanno Millesi, Ludwig Boltzmann Institut für E perimentelle Plastische Chirurgie.

DYNAMIC MODELS OF LATVIA RURAL COMMUNITIES

I. Ruža
Latvia University of Agriculture
Liela iela 2, Jelgava, LV-3002, Latvia
e-mail: inguna@cs.llu.lv

National and international authorities must make difficult policy decisions regarding socio-economic problems, which are complex, highly interrelated, and subject to uncertainty and external disturbances. Analytical and simulation models have proven useful in helping decision-makers to understand the processes involved in these complex problem/policy contexts.

The aim of study is to investigate the behaviour of communities, the factors what influence changes in this system and these changes under different local government and state policies. Community system is complex and its state today depends from yesterday's policies. It is obvious that not all well-thought policies rich the goal. In fact - most of them do not bring desired changes and some of them even make situation worse.

Latvia's sore point is rural regions. Young and educated people are leaving home villages and moving to urban regions. Rural area becomes older, land is abandoned and from the other hand unemployment rises. To show these dependencies and casuals is used the theories of Systems Thinking and System Dynamics

System dynamics methodology uses computers' simulation models to relate the structure of a system to its behaviour over time. It is a non-linear, dynamic, feedback-based technique that is able to portray system behaviour as it actually occurs – i.e., in disequilibrium, with decisions being made by imperfect humans using imperfect information. System dynamics models are powerful tools to help understand and leverage the feedback interrelationships of complex management systems. The models offer an operational methodology to support decision-making. Decision-makers can use the models to test "what-if" scenarios and explore what might have happened – or what could happen – under a variety of different past and future assumptions and across alternative decision choices. Regional development requires a system-oriented treatment.

To create the dynamic model of a rural community for simulating its development over time is set five levels – Population, Local budget, Welfare, Capital and Land. Each of these levels represents the principal variable in a major subsystem of community structure.

For developing simulation model is important to interconnect all these different sectors of a region and reflect their cross linkage, interfacing in reality and their feedback behaviour. To make a basis of the whole model are made several assumptions: community is closed system hence it is possible to select main factors what influence behaviour of the system; community is small part of state so there is not independent problem of food self-sufficiency and pollution and hence we omit them; the main recourse is arable land and main capital - agricultural capital. There are such main variables: population shift, population economical structure, available land resources, land for agriculture uses, community budget division, income per capita, employment, industrial capital, service capital, agriculture capital, trade flows, etc.

For the basic model are chosen average characteristics of parameters. The problem in creating the model is lack of statistical data from near past on the subject of political changes in our country so some of parameters have to be estimated. From this basic model are not expected to get exact prediction of different policies outcome nor exact numbers about population, capital, etc. in concrete year. It is made for increasing knowledge about system behaviour mode under different circumstances. It is made as base what is possible adapt to create concrete communities models.

The model is created using software – Powersim Constructor.

References

Ruža, I., Rural communities decision making based on dynamic models. In: Proc. Rural integrated and sustainable development strategy: problems, models and key actions, seminar, Lithuania, 1999, Kaunas.

Ruža, I., System Approach in rural community developing projection. In: 196th Transactions of the Estonian agricultural university. Tartu 1998, 179 - 181.

Comparison of implementation models for fuzzy relations

J. Scheikl, F. Breitenecker, M. Wibmer, Dept. Simulation, Vienna Univ: of Technology

Since its development in 1965 the theory of fuzzy sets and fuzzy logic has become an acknowledged method of engineering. Especially the use of fuzzy controllers is wide spread. Fuzzy logic quickly found its way into control engineering and thereby into modelling and simulation of control systems.

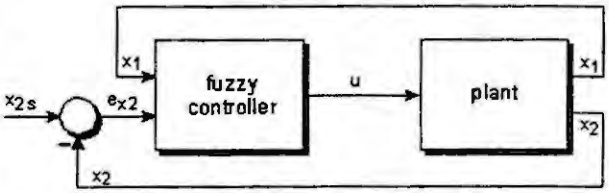
A controller can be described as an *m*-dimensional function of an *n*-dimensional input parameter, which means that the controller has *n* inputs and *m* outputs. If we describe the relation between the inputs and the outputs by means of fuzzy logic, we will get a fuzzy function (fig. 1). Because fuzzy functions like the one described above have up to now been used mainly in control engineering, they are normally called controllers.

Fig. 1 Steps of Fuzzy Relation

The evaluation of a fuzzy function is done in three steps according to the membership functions of the linguistic variables (fuzzy sets) and the operations defined fuzzification, interference, defuzzification (fig. 1).

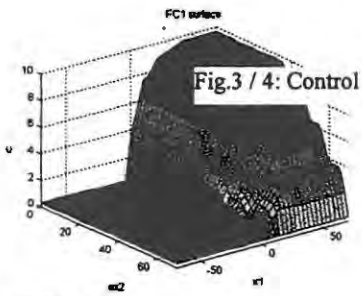
ARGESIM features a series on comparisons of simulation software in the journal Simulation News Europe (SNE) and on the WWW. Based on simple, easily comprehensible models special features of modelling and experimentation within simulation languages are compared. Comparison 9 (Fuzzy Control of a Two Tank System, fig. 2) asks for modules for fuzzy control or how such modules can be implemented efficiently.

Fig.2: Fuzzy Control in C9 – Two Tank System



Up to now 11 solutions have been sent in, showing all differences in the quantitative behaviour in the transient (but resulting in the same stationary value in nearly equal times (fig.3 control surface, fig. 4 states and control)

At present an evaluation of the comparison solutions is under investigation, which tries to find the reason for the different transient behaviour – classifying different implementation models for fuzzy modules.



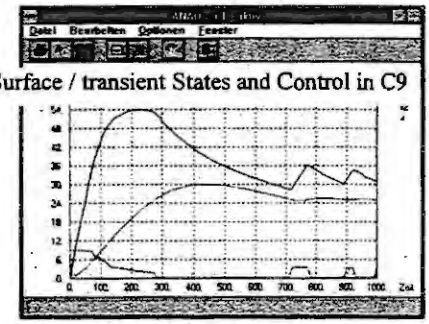


Fig.3 / 4: Control Surface / transient States and Control in C9

In these 11 solutions

i) either modules for fuzzy control were used, which support fuzzification, inferenz and defuzzification by means of predefined algorithms - ***numerical model*** -

ii) or the fuzzy control was programmed directly (because of the lack of a fuzzy module) – ***analytical model*** –

While solutions type ii) showed nearly the same transient behaviour, solutions type i) showed a different one, also within this type.

Careful investigation found a possible reason: fuzzification and defuzzification is calculated "purely" numerically by using a grid discretisation for representing membership functions. Intermediate values were calculated by interpolation.

This method may fit well for membership functions of e.g. Gaussian curve type, but in case of the widely used rectangular, triangular or trapezoidal membership functions, these grids cause large interpolation errors. Fig. 5 shows the interpolation of a membership function p1 by a too big grid, resulting in the dashed line.

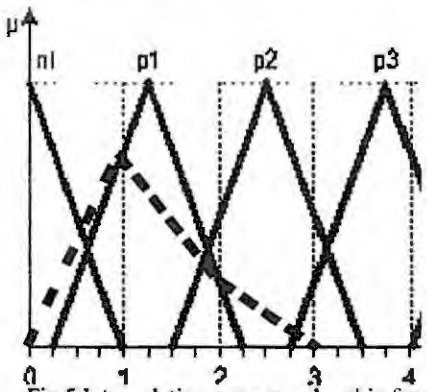


Fig.5:Interpolation error membership function

As not all phenomena could be made clear with the interpolation error in fuzzification and defuzzification, further investigation were necessary. A comparison of the simulation times brought some clearness: some simulations in the time domain (fig.4) were very fast, faster than the calculation of the control surface (fig.3). As the solution of the differential equations together with calculation of the control must take more time than the calculation of the surface, these solutions must work in simulation with a precalculated control surface, so again two kind of implementation models can be found:

a) ***Algorithmic fuzzy model***: the new control is calculated on demand by fuzzification, inferenz, and defuzzification

b) ***Fuzzy data model***: After definition of the fuzzy control and setting of maximal values for the inputs a control surface is calculated in advance. A new control is interpolated from the control surface.

It is quite interesting, that the solutions sent in used different mixes of the implementation models, which all have advantages, but also disadvantages (Table 1). Improvements for the future could be made by means of symbolic approaches.

	analytical model	**numerical model**
algorithmic model	nearly exact, slow, to be programmed	medium interpolation error, not to slow, modules available
data model	medium interpolation error, fast, to be programmed	large interpolation error, very fast, modules available

Table 1: advantages / disadvantages of implementation models

A GEOMETRIC MODEL FOR COMPUTER-BASED ANALYSIS OF BAROQUE MINIATURES

J.Scheikl[1], N. Popper[1], P.Kammerer[2], E. Zolda[2]

[1]ARGESIM-Dept.. Simulation, Vienna Technical University, Wiedner Hauptstr. 8-10, A-1040 Wien
[2]Patter Recognition & Image Processing Group, Vienna Technical University, Treitlstr. 3, A-1040 Wien
joxg@osiris.tuwien.ac.at

PAINTING TRADITION IN THE 18TH CENTURY

In Baroque Times the Artists learned their ‚profession' according to strict rules. Hereafter they added their personal style and art! For the depiction of human heads the rules demanded ellipsoid alike representations. Within this model the location of eyes, ears, the mouth and nose, etc. was clearly specified.

Today this ellipsoid model can be retrieved from the pictures. The art historians investigating these paintings can use this reconstructed model for their analysis.

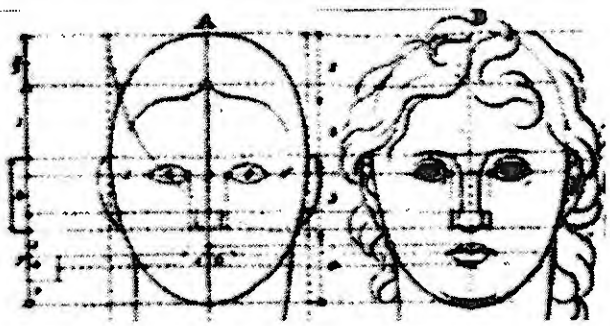


Figure 1: Painting Scheme

MODELLANSATZ

The model is based on the assumption that the artists painted according to the elliptical scheme. The portrait of the human head arises form a parallel projection into R2.

The location of the resulting ellipse is determined by some feature-points (chin, lower and upper end of ears, left and right nd of eyes etc.). These points also allow to determine the position of the depicted head in R3.

REKONSTRUKTION

The reconstruction of the ellipsoid model is based on *Datafitting*. In a first step the 'face-ellipse' is calculated.

Two methods are used: First a direct leas squares approximation second the analytical method described by Pilu [1]. The least squares method minimises the distance of a set of points on the contour of the face to a general ellipse. This can be done with sufficient results since a good starting value can be constructed. The direct analytical method is based on solving a generalised eigen system.

Both methods yield similar results where the analytical method is reasonably faster.

Figure 2: Face Ellipse

3-D MODEL OF THE PORTRAITS

Assuming that the face model is based on a rotating ellipse the 3D model can be obtained. For that certain feature points are used. The nose always lies in the vertical symmetry plain, top end of ears and left and right end of eyes can be found on an horizontal plain.

The resulting 3D model allows to extract specific regions of the paintings. For example one might want to compare areas of the paintings which show a right cheek which points towards the viewer. These areas can be extracted of all paintings and analysed by the art historian.

LITERATUR

[1] M. Pilu; A.W. Fitzgibbon; R.B. Fisher;"Ellipse-Specific Direct Least-Square Fitting; ICPR; 1996; Vol. A; S. 253-257

This project is partly funded by the Austrian Science Fund under grant P12028-MAT.

SIGNAL COMPUTING IN SMALL BIOLOGICAL NEURAL NETWORKS

P. Slowik, L. Mehnen and F. Rattay

TU-BioMed, Vienna University of Technology, Austria

slowik@mail.zserv.tuwien.ac.at

Simulating the flow of information in neural networks is a novel tool for naturally and artificially evoked spike trains. The prediction of neural activities generated via implanted electrodes are of highest clinical interest, e.g. for sensory prostheses, spinal cord stimulation or deep brain stimulation in case of Parkinsons disease.

Introduction

The high degree of neural interconnections within the central nervous system has been a major obstacle for the development of signal flow theories.

Material and Methods.

Our NeuronNet simulator was built to handle problems of temporal coding in biological neuron structures. The NeuronNet is based on interconnected model neurons, the information is transported in form of electrical potentials. Every neuron consists of four components: dendrites plus soma, initial segment, axonal pathways and synapses (see Fig 1). Specific 'synapses' represent the natural or artificial input. For every component of the system, the reaction is simulated as a delayed and weighted function of the input signals. Inhibitory effects are simulated by a time dependent shift of the signal amplitude.

The user has to design the neuron - structure (geometric and electric parameters) and the input data (neural impulses specified according to location, time and amplitude) of the NeuronNet, both acting as program and data simultaneously. Due to this fact, the classical way of linear programming is not possible any more. The program / data has to be established as "a whole and in action", because every single part acts in parallel. The combination with other simulators, e.g. to model nerve-muscle interactions is of special interest for applications in functional electrical stimulation, but it has to be taken care of the parallel character of the data. Other fields of application are retina or cochlear implants. Action potentials can be generated or blocked in excitable tissue by extracellular electrodes or magnetic coils. New results about artificial excitation are based on a compartment model and on the theory of the generalized activating function.

Results

The NeuronNet was tested on a spinal cord reflex arc consisting of 36 neurons and a small retina neural network model. The simulated firing patterns are comparable with experimental results. The present model for the soma does not account for individual ion gating mechanisms that are responsible for specific properties like pacemaker activities. However, such properties can be included in the model system by introduction of additional neurons.

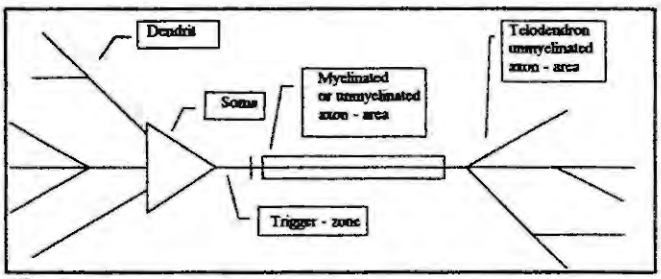


Fig. 1

Conclusion

Artificial neural networks, which have some similarities to fundamental structures in the brain, are highly established in many technical applications. Nevertheless, there are some essential features in the biological system that cannot be modeled with the conventional techniques. Adding a local / temporal component results in a fundamental redesign, yielding in NeuronNet, a novel computational tool that simulates spiking actions in a biological neural network.

References

P. Slowik 1998 Informationsfluß in simulierten Neuronensystemen, Dissertation TU Wien

P. Slowik 1999. Informationsflußanalyse in komplexen Neuronensystemen. ÖGAI 3/1999, 10 - 17.

F. Rattay, P. Slowik and L. Mehnen 1999. Signaling in small biological networks: a modeling study of naturally and artificially evoked spiking patterns. Med. & Biol. Eng. & Comp.37, Suppl. 2, 1206-1207

MATHEMATICAL MODELLING OF THE TECHNOLOGICAL PROCESSES TREATED AS DISTRIBUTED PARAMETER DYNAMIC SYSTEMS.

Nasta Tanasescu*, Adrian Filipescu*, Ovidiu Tanasescu°

**University "Dunarea de Jos" of Galati, Department of Automation and Electronics,*
Domneasca Street,no.47, 6200, Galati, Romania, Fax: 40 36 46 01 82,
e-mail:nasta@linux.ac.ugal.ro
°Student "Politehnica" University, Bucharest, Romania

Abstract: The paper presents the latest researches on the modelling of the Distributed Parameter Systems and the simplification of these models in order to design the automatic control systems. The description by transfer matrices having complex variable Laplace as transcendent functions is regarded by the authors as the most complex and significant.
The process mathematical model can be obtained under two representations: a non-parametric representation, as for example (Bode or frequency characteristics diagrams [Sidman, M., D., et al., 1991]) and a parametric representation as a transfer function multiplied by a transfer function of a dead time element.

3. MATHEMATICAL MODELLING OF A HEATER REGARDED AS A DPS

3.1. Determining the heat balance and material balance equations.

a. The stream balance equation

$$T_1\frac{\partial\theta_1(\xi,t)}{\partial t}+Y_1\frac{\partial\theta_1(\xi,t)}{\partial\xi}=\theta_m(\xi,t)-\theta_1(\xi,t)+Kw(\xi,t) \qquad (4)$$

b. The metal balance equation

$$\frac{\partial\theta_m(\xi,t)}{\partial t}=q(t)+\frac{1}{T_m}[\theta_1(\xi,t)-\theta_m(\xi,t)]-\frac{m}{1-m}Kw(\xi,t) \qquad (5)$$

c. The model output equation is a material balance equation and connects the velocity to the state variables

$$b\frac{\partial w(\xi,t)}{\partial\xi}+dw(\xi,t)=-c\frac{\partial\theta_1(\xi,t)}{\partial\xi}+a\frac{\partial\theta(\xi,t)}{\partial t}-e\theta(\xi,t) \qquad (6)$$

The parameters to be found in the heater model given by the equations from (4) to (10) have the following meaning:- θ_1, θ_m, q, w, p are the fluid temperature, the metal temperature, the heat inflow density, the velocity and the fluid pressure, respectively; T_1, Y_1, K, T_m, m, a, b, c, d, e, a_1, a_2, k_1, k_2, k_3, α_1 and α_2 are constants calculated depending on the heater constructive characteristics and the nominal values of the physical parameters.

The heater described by the above equations can be regarded as a multi - variable system.

$$\theta_1(\xi,s)=H_1(s)\cdot\theta_o(s)+H_2(s).w_o(s)+H_3(s).q(s)$$

Similarly the output equation w(L, s) processed

$w(L,s)=M_1(s)\theta+M_2(s).q(s)+M_3(s).w_O(s)$ where, for example:

$$M_1(s)=\frac{1}{p_1(s)b+d}[cp_1(s)+as-e]f_{11}(s)e^{\left[p_1(s)+\frac{d}{b}\right]L}+\frac{1}{p_2(s)b+d}[-cp_2(s)+as-e]f_{12}(s)e^{\left[p_2(s)+\frac{d}{b}\right]L} \qquad (25)$$

Table 1

	Non parametric model	Parametric model
	$H_1(s)$ - equation (21)	$h_1(s)=\frac{0.2+2.123s}{1+0.69299s+0.0228s^2}$
	$H_2(s)$ - equation (22)	$h_2(s)=\frac{-0.1173-0.0162s}{1+0.5124s+0.14s^2}$
	$H_3(s)$ - equation (23)	$h_3(s)=\frac{-0.696+0.1185s}{1+0.5226s+0.465s^2}e^{-0.4336s}$
	$M_1(s)$ -equation (25)	$m_1(s)=\frac{-0.0584+0.208s+0.4397s^2}{1+0.1164+0.095s^2}e^{-0.5195s}$

1

Expert System for Vascular Surgery based on a Stationary Blood Flow Model

S. Wassertheurer[1], Ch. Almeder[1], F.Breitenecker[1], J. Krocza[2], M. Suda[2]
[1]Technical University of Vienna
Wiedner Hauptstr. 8-10, A-1040 Vienna, Austria
[2]Austrian Research Centers Seibersdorf
A-2444 Seibersdorf, Austria

Abstract. The aim of this project was the development of a user-friendly software package for physicians that can be used as an advisor in vascular surgery and as a training tool for medical students. The program consists of a graphical user interface, a mathematical model that describes the relationship of morphology and hydraulics in human arterial networks and an expert system for managing automatic parameter identification and bypass optimization.

INTRODUCTION

The main purpose of the model of the human arterial network is to describe the relationship of the morphology and the hydraulics. This model offers methods to calculate mean flow velocity, mean flow, flow direction and blood pressure in the arteries.

MODEL DESCRIPTION

Graph theory is used to get a mathematical representation of the network topology. The pipe network is translated into a directed graph and the structures is stored in a node-edge-incidence matrix.
The nonlinear hydraulic equations are based on three conditions (*node, mesh , hydraulic condition),* which lead to a system of non-linear equations. The number of unknowns depends on the number of nodes in the network (usually between 150 and 300). In most cases when a arterial system is modeled, only laminar flows occur in the whole network. Then the resulting equation system is linear and sparse. If turbulent flows or flows in the transition range occur, the solution of the system is approximated using a fix-point iteration. [1,2]

PROGRAM STRUCTURE

The implementation of the mathematical model is the basis of the software package, but some other modules are necessary so that the program can be used easily.

1. **Expert System**
   The expert system controls the mathematical module. It automates the parameter identification and adapts the standard networks to measurements. Another task of the expert system is the processing of simulation experiments. So automatic bypass optimizations can be performed to find an optimal operation method.
2. **User Interface:**
   Our prime directive in user interface development was to implement a workplace that allows complete intuitive software usage, even the user is a "Non-Computer-Expert".
   The user interface provides two main parts: a patient database, which is designed to manage the personal data and the different vessel models, and a graphical editor for supporting a corresponding visual description of real live vessel systems. [3]

CONCLUSIONS

The presented software realization of the model is already in use by physicians for scientific investigations. Those tests have shown that a good prediction is possibly based on few flow and pressure measurements.

REFERENCES

[1] Ch. Almeder. 1997. *Simulation of the Human Arterial System.* Diploma thesis, Technical University of Vienna.
[2] M. Suda, O. Eder, B. Kunsch, D. Magometschnigg, H. Magometschnigg. 1993. *Preoperative assessment and prediction of postoperative results in an artificial network using computer simulation.* Computer Methods and Programs in Biomedicine. 41:77-87.
[3] E. Angel. 1997. *Interactive Computer Graphics.* Addison Wesley.

The Concept of Extended Data Models for Representation of Simulation Experiments

M.Wibmer*, J.Scheikl*, F. Breitenecker*
P. Krejsa**, R.Rybin**
*Vienna University of Technology, Wiedner Hauptstrasse 8-10, 1040 Vienna, Austria
**Austrian Research Centers, A-2444 Seibersdorf

Within simulation experiments it is often necessary to obtain the solution of the underlying differential equation between mesh points. This is particularly significant when, as a part of a longer calculation, the solution of the problem is required at various locations. Important examples are the use of an automatic plotter tbat frequently requires interpolation at a many intermediate points and the development of a simulation environment which is useable in computernetworks, especially the World Wide Web. Such environments must therefore be implemented in a client-server architecture. Most of the simulation environments, such as MATLAB/SIMULINK or ACSL cannot be used witbin a client-server simulation environment because of their license politics. So one has to search for a simulation language independent representation of the solution of a simulation experiment. One concept is the Fieldmodel [1], which provides a discrete representation of the dependent variable space of differential equations. Due to interpolation one can achieve a continous representation of the dependent variable space. The Fieldmodel construct a representation of numerical data resulting from the solution $x(x_0, t, p)$ of an ordinary differential equation $\dot{x}(t) = f(x, p), \quad x \in \mathbb{R}^n, \quad p \in \mathbb{R}^m$ with inital conditions $x(0) = x_0$. The definition of a Fieldmodel is as follows:

A Fieldmodel space S is the cartesian product of the basis manifold Ω and the set of dependent variables Y,

$$S = \Omega \times Y,$$

and is defined by adding a mapping $\Phi : \omega \to \Omega$, where $\omega \in \mathbb{R}^k$ and a description for Y. A Fieldmodel F is a pair of mapping

$$F = (\Phi, \Psi),$$

which share the same domain ω of Φ.

The Fieldmodel reflects the structure of the function, which gives the datapoints. It is also constructed to reflect the dynamic behavior of the differential equations.
The Fieldmodel is implemented in a simulation environment, called WSim V1.0. WSim is implemented in JAVA 1.2. If one bas calculated a discrete representation of the solution of the underlying differential equation, including solutions for different values of the parameter vector p, the datapoints can be stored in an Oracle database with help of WSim V1.0. WSim then calculates the Fieldmodel F. This Fieldmodel can now for example be used in the field of visualisation. WSim V1.0 uses the JAVA-package VisAD, a tool for visualisation of numerical data. WSim V1.0 provides a JAVA-Applet, so one can simulate the existing models over the World Wide Web.
This work is only a first attempt for the efficient representation of solutions of ordinary differential equations. In future work it will be useful to introduce approximation methods in the field of simulation to have an analytic representation of the solution, i.e. with help of taylor series expansion or approximation by splines.

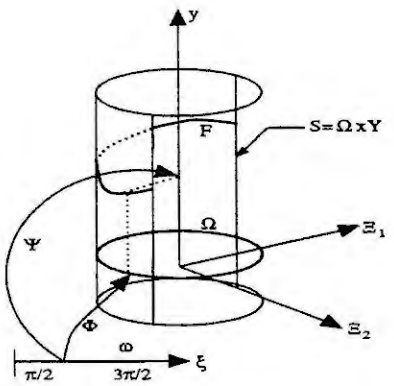


Figure 1: Structure of a simple Fieldmodel

References

[1] Haber R.B., Lucas B., Collins N., *A data model for scientific visualization with provisions for regular and irregular grids*. Proc. Visualization 91, IEEE, pp. 298-305.

Modelling of the Seasonal Snow Cover

Karl Kleemayr, Sigrid Wieshofer
University of Agricultural Sciences, Institute of Torrent and Avalanche Control
Peter Jordanstrasse 82, A-1190 Vienna

Abstract

In this short paper we present a finite element model to calculate the forces and displacements in the snowcover and the problems occurying within this context shall be announced.

Introduction

The objective of the investigations is to develop a tool which enables to quantify the movements, deformations and stresses of the seasonal snow cover. This is of practical importance in order to optimise technical supporting structures to control and prevent avalanches, to analyse the influence of the soil roughness and the conditions in the snow-ground interface layer and to analyse the stabilising effect of mountain forests and technical supporting structures.

Problem Description

The system study distinguishes two movement components in a snow cover, creep (internal deformation), dependening on gravity load and snow parameter as well as snow gliding (gliding of the entire snow cover), caused by factors such as free water in the junction region and lack of macroscopic roughness. The formation of a snow cover model, above all the development of an universal material model is difficult on account of the multiple appearances of snow. The snow model is based on the simplification that the snow cover is an isotropic and homogenous material whereby the weight of snow is the only factor acting from the outside on the system. The calculations and analysis had been done with ABAQUS which is a standard Finite Element program used for engineering applications. Using continuum elements, the assumption of a continuous medium for the snow cover could be realised. Two components were relevant for the modelling with finite elements: a two dimensional (plain strain elements) model has been established to determine the effect of soil roughness onto the displacement of the snow cover. A three dimensional model has been made to analyse the spatial stress distribution within the snow cover and to get information on how the tree population and also technical constructions interact with the snow. In the first place the interaction between snow cover and ground surface was defined by the Coloumb dry friction law. To include drag forces caused by macroscopic ground roughness the surfaces was modelled by a sinusoidal wave with respect to McClung.

Conclusion

The comparison of the simulation results, obtained by the two-dimensional model through variation of the soil wavelength and amplitude, showed that increasing snow depth does not necessarily lead to higher gliding velocities because the interlinking of the snow cover with the ground surface affects higher normal resistance forces.

References

[1] David M. McClung, *Creep and Glide Processes in Mountain Snowpacks* , Nat. Hydr. Research Institute, 1980

Method SPAdd for the decision of inverse tasks on models the sustainability development.

Zagoruiko N.G..
Institute of Mathematics SB RAS
Novosibirsk, Russia. E-mail: zag@math.nsc.ru

The models intended for search the strategy of transitions in system of sustainable development, differ by a high degree of complexity. So, subsystem, developed by us, "POPULATION" simulating the transition processes of human resources, has 9 blocks and 31 control parameters, which determine influence of blocks against each other. The direct task (task of the analysis) consists in imitation of processes in system at the given values of control parameters. The inverse task (task of synthesis) consists in search the values of control parameters ( ), which would provide the given quality (Q) the functioning of system. For the exact decision such np-complete problem 10 in 14 degrees of years working the modern computer would be required.

The method directed look through (SPAdd), using the modified methods of random search with adaptation (SPA) and consecutive choice (Add) is described. In a method SPA the choice n most informative parameters from initial set q is carried out. In the beginning all q parameters have equal probability to be selected for optimum combination. By a random way gets out n of parameters and the function of quality it combination Q is estimated. Such random combinations get out and are estimated r times. Then the procedure of adaptation is carried out the parameters including in structure of the best combination, "are encouraged" (probability of their choice is increased), and the parameters from the most unsuccessful combination "are punished" (probability of their choice decreases). Such cycles of a random choice and adaptations repeat up to one of stopper criteria. In result a decision obtained usually close to optimum. In our case the set of control parameters does not vary and it is necessary to adapt a method SPA for searching a good combination of their values. The method of a consecutive choice Add consists in a serial rating the influence of all control parameters on one. Most influential parameter is fixed, and all other parameters join it on turn, therefore there is a most influential pair parameters. It is fixed and by the same way third most influential parameter joins. The procedure repeats up to a choice of the given number the most essential parameters. In our case it is necessary to choose and to estimate not sets of parameters, but combination of values of their given set. The association the opportunities of these two algorithms has allowed creating an effective method the decision of inverse tasks in models of complex systems.

References:

1. Lbov G.S. The Selection effective systems depending features. Proc. of Institute Mathematics SB RAS "Computer Systems", Novosibirsk, 1965. Vol.19 pp. 21-34.
2. Barabash Yu.L. et. al. Automatic Pattern Recognition. Edition KVAIU, Kiev, 1963.

The work was supported by the Grant RFFI 99-01-00582 and Grant MinSc PIT-0201.05.238

DETERMINATION OF AR PART'S ORDER (p_1, p_2) OF A 2-D ARMA $(p_1, p_2; q_1, q_2)$ MODEL

B. Aksasse, L. Badidi, and L. Radouane
LESSI, Departement de Physique, Faculte des Sciences,
B. P. 1796 Atlas Fes, 30000 Morocco
E. mail: baksasse@yahoo.com

Abstract. In this paper, we make some investigation on two-dimensional autoregressive moving average (2-D ARMA) model order estimation problem. As the AR order (p_1, p_2) is usually of more practical importance compared with the MA order (q_1, q_2). We will now concern ourselves only with resolving this crucial problem of estimating (p_1, p_2). We develop an approach to estimate the (p_1, p_2) order via the SVD method. The algorithm is based on the computation of three correlation matrix ranks. We will show that these ranks implicitly contain the information on the (p_1, p_2) order.

1. Introduction

Random field models have been successfully utilized in many applications for image processing and analysis, requiring: 2-D spectral estimation [3], [8], [14], [16], image synthesis, classification and modeling [10], [11], [12], image enhancement and 2-D Kalman filtering [7], [13]. The problem of model order selection is of utmost importance. For the purpose of analysis, it is assumed that the data $y(i, j)$ can be modeled by a stable and minimum phase 2-D ARMA model of the form [3]

$$\sum_{k_1=0}^{p_1} \sum_{k_2=0}^{p_2} a_{k_1,k_2}\, y(i-k_1, j-k_2) = \sum_{k_1=0}^{q_1} \sum_{k_2}^{q_2} b_{k_1,k_2}\, e(i-k_1, j-k_2) \qquad (1)$$

where $\{e(i, j)\}$ is a 2-D Gaussian white noise sequence whose elements have zero mean and variance σ_e^2. $\{a_{k_1,k_2}\}$, $\{b_{k_1,k_2}\}$, and $(p_1, p_2; q_1, q_2)$ stand for the AR parameters, the MA parameters, and the model's order, respectively. It is widely appreciated that the ARMA model of order $(p_1, p_2; q_1, q_2)$ as specified by equation (1), is generally the most effective quarter plane (QP) causal and linear model. From a parameter parsimony viewpoint, this general 2-D ARMA model usually provides the most effective linear model of a homogeneous field and is therefore preferable over its 2-D AR model counterpoint [3]. In practice, most applications employing 2-D AR and ARMA parametric models assume that the model's order is known a priori or rather choose an arbitrary order. However, this model's order is not known and is a crucial problem. In some situations, we have only interest on 2-D AR parameter [3], [14], [16]. In [3] e.g., Cadzow and Ogino have developed an effective method for generating 2-D QP-ARMA spectral estimation model. This method consists specifically in two steps:

- A spatial domain approach, which it consists on minimizing a model error criterion for estimating the ARMA model's AR parameter values.
- This is turn followed by a frequency domain approach of estimating the effects of the MA coefficients on the overall spectral estimate. Which is done by the use of the well-known *Welch* method for obtaining smoothed periodogram estimate.

Therefore, the estimation of (p_1, p_2) is enough to estimate the ARMA model's AR coefficient values. From one-dimensional (1-D) time series analysis, it is well known that the use of an appropriate model leads to good results in forecasting, process control or spectral estimation. A lower or upper model's order affects heavily the result quality. An upper model's order adds to the computational burden can potentially degrade the overall performance of the model as an additional source of noise. Moreover, the problem of model-order determination has received considerable attention during the past decades, which has resulted in a lot of order determination approaches. These approaches can broadly classified into two main categories, namely, information theoretic criterion (ITC) approaches and linear algebraic (LA) methods. The AIC criterion [1] developed by Akaike and MDL criterion [9] of Rissanen are well examples for the first class. The determinant testing algorithm of Chow [6] and the Singular Value Decomposition (SVD) of Cadzow *et al.* [4], [5], [15] are well examples for the second category.

SIMULATION OF COMPLEX PIPE SYSTEMS AS THE SYSTEMS WITH LUMPED PARAMETERS

Arkady A. Atavin[1] and Vladimir V. Tarasevich[2],
[1]Institute for Water and Environmental Problems
Papaninzev str., 105, Barnaul, 659099, Russia.
[2]Novosibirsk State University of Architecture and Civil Engineering (NGASU)
Leningradskaya str., 113, Novosibirsk, 630008, Russia

Abstract. The large complex piping systems under unsteady operating mode are considered. Such systems can exemplify the typical systems with distributed parameters. General mathematical model is demonstrated. The process on the base of spatial averaging, for replacement the system with distributed parameter by the system with lumped parameters is described. The example of specification of this process for the piping system is given. The system with lumped parameters consisting from inertial and elastic blocks are used for substitution the system with distributed parameters. The procedure of system convolution into the node is described. The comparison "thorough" system with simplified one was realized for some models.

Introduction

Many widespread systems with the distributed parameters (for example, network of water supply, thermal networks etc.) contain the thousands of elements and have very large sizes. The calculation of such systems requires significant computational resources and occupies a rather long time. But many problems arising in practice, for example, the problem of on-line control, require fast calculation and forecast of a system state.

Increase of computer performance and the improvement of computational algorithms, results in reduction of computational time of course. However the creation of "fast" simplified models seems more perspective way.

One of main tendencies for creation such simplified models consists in replacement the "original" systems with the distributed parameters by the systems with the lumped parameters. The problem is to match the characteristics of this new system so that parameters of process in it were "nearer" to parameters of process in basic simulated system as possible.

Mathematical formulation of problem in common case

The notion "system" will mean some set of "elements" and "relations" between them: $S=\{M,R\}$, where M is the set of elements, R is the set of binary relations ("links") between the elements. Let consider the system where the set of elements may be divided into two subsets: $M=C\cup N$, where C is the set of currents in the system; N is the set of system nodes (i.e. the points of junction, branching, transformation of flows etc.).

The system structure can be described by graph Γ, whose vertexes correspond to nodes (N) and arcs correspond to currents (C). Assume that k is index (subscript) of arc and j is index (superscript) of vertex.

The flow parameters are described by vector $\vec{u}_k(x,t)$ where x is spatial variable and t is time, i.e. the elements of C are the subsystems with distributed parameters. The flow parameters should satisfy the equation

$$\vec{F}_k(\vec{u}_k, x, t) = 0\,, \text{ where } 0 \le x \le L_k\,, \tag{1}$$

describing the current along the arc, where L_k is the k-th arc length. Here the operator $\vec{F}_k$ represents by itself the system of partial differential equations with respect to unknowns $\vec{u}_k(x,t)$.

Each node is simulated by the system with lumped parameters:

$$\vec{H}^j(\vec{u}^j, \vec{\Lambda}^j, t) = 0\,, \tag{2}$$

where $\vec{H}^j$ is the system of ordinary differential equations and/or algebraic equations, $\vec{\Lambda}^j$ is the vector of intrinsic node parameters, $\vec{u}^j$ is the vector composed of all parameters of input flows and exit streams at node.

It is necessary to assign the initial data under $t=0$ for the unambiguous solution:

$$\vec{u}_k(x,0) = \vec{u}_{0,k}(x),\ \ \vec{\Lambda}^j(0) = \vec{\Lambda}^j_0 \tag{3}$$

Thus the problem of flow modeling is reduced to the so-called graph-defined initial-boundary value problem [4] for the system (I) with the boundary conditions (2) and the initial data (3).

The solution of this problem by computer requires significant computational resources, especially for large and complex systems containing the thousands of elements. The procedure of calculation acceleration is a

PARALLEL AND DISTRIBUTED STATE - SPACE MODELING FOR COMPUTATION OF TIME SERIES USING REALIZATION THEORY

Celso Pascoli Bottura, Gilmar Barreto, Maurício José Bordon & Annabell del Real Tamariz
School of Electrical and Computer Engineering - State University of Campinas, UNICAMP,
Caixa Postal: 6101 , cep : 13083-970, Campinas, Brazil, gbarreto@fee.unicamp.br.

Abstract. Based on previous works by Kalman, Akaike, Aoki, Verhaegen, Dewilde, De Moor and Van Overschee, among others, the objective of this paper is to present an algorithm for state-space modeling of time series using realization theory for linear discrete-time systems. By exploring the matrix structure for the state-space model via basic techniques of numeric analysis: Householder rotations, Givens rotations, **QR** decomposition and singular value decomposition, a parallelized and distributed computation proposal for the the algorithm is presented.

Introduction

A central problem in systems analysis and design including, for example, economic systems analysis and control, is state space modelling of time series based on multiple-output data sequences.

An approach to this problem is to determine an association between output data sequences and state space model of unknown parameters.

In a proposal by Aoki, the parameters estimation problem is solved through an Algebraic Riccati Equation solution. Such an approach to this problem yields strong computational burden.

In this work, inspired on Aoki's and on Verhaegen and Dewilde's algorithms we propose an alternative algorithm for linear time invariant state space modelling of multivariate time series. The main characteristic of Verhaegen and Dewilde's, (1992), method is an approximation of a vetor subspace defined by the column and row spaces of matrices constructed from system state space model matrices. This approximation is made through output data samples. The linear time invariant state space model is calculated from an approximate knowledge of these subspaces. Aoki's method benefits from the formulation of an optimal filtering problem for the calculation of the input matrix.

State Space Method For Time Series

State space modeling of time series consists in determining a realization of a dynamic system which posseses similar dynamic behavior to an implicit model we suppose is generating a stochastic process $\mathbf{y}(1)$, $\mathbf{y}(2)$, $\mathbf{y}(3)$, ...

We are interested in the construction of a linear time invariant stochastic dynamical system of the form :

$$\mathbf{x}(t+1) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{e}(t) \tag{1}$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{e}(t) \tag{2}$$

where :

- $\mathbf{y}(t) \in R^l$ is the output vector;
- $\mathbf{x}(t) \in R^n$ is the state vector;
- $\mathbf{e}(t) \in R^p$ is white noise vector.

In order to calculate matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}$, only $\mathbf{y}(t)$ is available and the white noise vector $\mathbf{e}(t)$ must satisfy the following conditions :

$$E\mathbf{e}(t) = 0 \tag{3}$$

$$E\mathbf{e}(t)\mathbf{e}(t)^T = \delta_{t,s}\Delta, \quad \Delta > 0 \tag{4}$$

The only information available is the measured sequence $\mathbf{y}(1)$, $\mathbf{y}(2)$, $\mathbf{y}(3)$, ..., $\mathbf{y}(N)$, of the weakly stationary stochastic process. Under this reasoning the innovation model we are intending to construct

Single-Stage Linear Approach for Fitting Motion Parameters of 3-D Point Sets

B. CHAOUKI, Lh. MASMOUDI AND L. RADOUANE
LESSI DEPARTEMENT DE PHYSIQUE
FACULTE DES SCIENCES BP. 1796
ATLAS-FES 30000 MOROCCO
e-mail: chaouki_bk@hotmail.com

ABSTRACT—The problem of estimation 3-D rigid motion from point correspondences over two views has been formulated as a linear least square solution. In the classical approaches, the unknown transformation (Rotation matrix and translation vector) is obtained in a two-stage linear approach. In this paper, based on the single-stage linear method for fitting rotation and translation parameters given two sets of 3-D points, a noise robust algorithm is presented. It takes into account the reliability of noisy measurement by introducing a weight matrix. The corresponding algorithm is noniterative and the unique solution is guaranteed. In this method, a set of intermediate parameters containing complete information about the unknown transformation is solved from measured data points. From these parameters, the rotation and translation can be uniquely determined. The simulation results show that the proposed solution is significantly more reliable than both the unweighted single-stage linear approach and the singular value decomposition (SVD) method which require two-stage.

1- INTRODUCTION

The estimation of three-dimensional parameters of a rigid body is an important problem in motion analysis. It can be useful in many applications such as scene analysis, motion prediction, target tracking, and so on. In general, to solve the problem requires the matching of two or three-dimensional data of feature points on the object at two time instances. After the matching of corresponding points has been accomplished, the motion parameters can be estimated by solving the equation, which governs the corresponding points at these two time instances. Linear least squares solution methods for solving the motion parameters were proposed in computer vision literature [1,2,3,4,5,6]. These approaches have a common feature; the unknown transformation is obtained in a two-stage process. The rotation part of the unknown transformation is solved in the first stage, using the two given sets of point measurements. The solved rotation, together with the same sets of point measurements, is then used to solve the translation part of the unknown transformation. On the other hand, Zhuang [7] was proposed a single stage linear method for fitting rotation and translation parameters given two sets of 3-D points. Unlike the existing approaches, six intermediate parameters are linearly computed in a single stage, using two sets of measurements. The rotation and the translation are then directly obtained from these intermediate parameters without reuse of the measured data. However, all these algorithms are constructed on the assumption that all the data are exact. Hence, they are all fragile in the sense that inconsistencies arise in the presence of noise, and the corresponding solutions are not optimal in that it equally trusts all components with different reliability. In this paper, a linear approach is proposed to solve the pose determination problem. Furthermore, the correlation between errors in the components of a three-dimensional point can be taken into account by using the covariance matrix as a simple weighting matrix. Indeed, due to the noise, it is reasonable to measure the reliability of each feature points. Reliable data have small covariance matrices and are thereby assigned large weights, whereas unreliable data have large covariance matrices and thereby assigned small weights.

III- A Single-Stage Linear approach

In the camera-centered coordinate system, we consider a set of corresponding three-dimensional points $\{P_i^1\}$ before motion and $\{P_i^2\}$ after motion, with $i = 1,2,..,m$. They are related by:

$$P_i^2 = RP_i^1 + T \tag{1}$$

where R is the rotation matrix and T is the translation vector. The 3D-3D pose determination problem is to infer R and T from the sets of corresponding points. From (1):

$$RP_i^1 = P_i^2 - T \tag{2}$$

Recall that $RP_i^1 = q \otimes P_i^1 \otimes q^*$ [8], where "$\otimes$"denotes a quaternion product and "*" denotes the Hamiltonian conjugate of the quaternion. Using this relationship, we obtain:

$$q \otimes P_i^1 \otimes q^* = P_i^2 - T \tag{3}$$

Equation (3) can be rewritten into:

$$q \otimes P_i^1 - \left(P_i^2 - T\right) \otimes q^* = 0 \tag{4}$$

Expanding (4) provides:

$$\left(q_{123} \cdot P_i^1, q_0 P_i^1 - P_i^1 \times q_{123}\right) = \left(q_{123}(P_i^2 - T), q_0(P_i^2 - T) - q_{123} \times (P_i^2 - T)\right) \tag{5}$$

MODELING THE CHEMICAL SYSTEMS FROM NEURAL NETWORKS

N. I. Korsunov and M. S. Rozanov
Belgorod State Technological Academy of Building Materials
Russia, 308012, Belgorod, Kostukov str., 46.
Email: root@intbel.ru

Abstract. The paper describes a method of modeling the chemical systems from neural networks. It means the determination of dependencies the properties of chemical system on some factors. Structure of modeling neural network and concrete algorithm of training are considered. After completion of training on quite power training set the behavior of chemical system is reflected by internal structure of neural network. It allows us to determine the properties of the system, depended on factors, which were not included into the training set. An example of modeling is described.

Introduction

Let the controlled factors of chemical system (e.g. composition, conditions of production) be described by vector X, and the properties corresponding to X by vector P. It may be supposed that some functional relations between factors and qualities are existed, i.e.

$$P = G(X). \tag{1}$$

Some set of pairs of vectors (X(i),P(i)), i=1,...,M is usually selected to build the mathematical model (Eq. 1) of the system, and on the base of this set the approximating functions $\tilde{G}$ for properties are determined, trying to minimize the error

$$E = \sum_{i=1}^{M} \left\| P(i) - \tilde{G}(X(i)) \right\|. \tag{2}$$

Tasks of this class belong to the problems of global multidimensional optimization. The complexity of its solving is explained as follows:

1. Formation the approximating functions (Eq. 1) is difficult in most cases, because chemical systems may be quite complex and its chemical and physical processes may be unknown.

2. Hypersurface of objective function (Eq. 2) may have a complex structure with a number of local minima, i.e. traditional algebraic approaches of minimization may be effectless.

One of the ways to solve the problem may be modeling such systems based on neural networks. This releases the experimenter from necessity of building the explicit models. Application the special methods of training the neural networks allows to avoid the problem of local minima of objective function (Eq. 2). Below the technique of building and training these networks is considered.

Collecting the experimental data

A set of pairs of corresponding vectors (X(i), P(i)), i=1,...,M is obtained by means of practical experimentation with a real chemical system. M is a power of training set. All vectors should be normalized, i.e. $X(i,j), P(i,j) \in [0,1]$ for all possible values of i and j.

To obtain the adequate results it is recommended to distribute the elements of training set over the space of factors. The more training components will be obtained the better neural network will model the chemical system.

Building the multilayer neural network

Based on the analyses of the problem's complexity, the structure of the network and links between its units are selected. General structure of L-layer neural network is shown at Fig. 1.

DIRECT MODEL REFERENCE ADAPTIVE CONTROL FOR NON MINIMUM PHASE CONTINUOUS TIME SYSTEMS

H. MEJHED AND L. RADOUANE
LESSI, DEPARTEMENT DE PHYSIQUE, FACULTE DES SCIENCES, BP 1796 - 30 000 FES-MAROC
Email: LESSI@rocketmail.com

Abstract. In this work, a direct model reference adaptive control algorithm for continuous time non-minimum phased systems is considered. First, we use an exponential input output data filtering to handle the non minimum phase assumption. Then, we introduce the delta operator to transform the continuous time (MRAC) into the discrete time one. The stability in the sense that the input and output of the plant are bounded is established. It is shown that the proposed data filtering permits also to reduce the effects of the unmodeled dynamics and noise. The performance of the control algorithm is illustrated by numerical and real examples.

Introduction.

The direct and indirect model reference adaptive control (MRAC) of linear continuous time systems has been successfully developed in the last decades. However, the various algorithms proposed in the literature are all based on the minimum phase assumption [1-11,14,18,20] and usually, introduce various modifications to enhance the robustness of the (MRAC) approach in presence of the unmodeled dynamics. The major available techniques include normalisation with parameter projection: The robustness of a direct (MRAC) with parameter projection is established in [3]. However, a prior knowledge on unmodeled dynamics is still required. In [10] the robustness is considered for a restricted class of unmodeled dynamics and extended in [9] by projection. Praly [18] introduced the device of using a normalising signal in parameter estimation. Recently, for the continuous time systems, the minimum phase assumption has been eliminated in [12] by relocating the poles and zeros of the controlled system by periodic feedback control with multirate sampling. The system is seen as a MISO plant. The stability and the convergence of the output error are established. However, the number of parameters to be estimated increases and sometimes it becomes so difficult to construct multirate-sampling systems for large orders. The problem in continuous time systems is generally, at fast sampling rates, the sampling zeros appear outside or on the unit circle [13,21], and then the model reference adaptive control (MRAC) method that involves cancellation of unstable zeros, cannot be used. To avoid such situation, the delta operator ($\delta = \frac{q-1}{\Delta}$ where q is the shift operator in the time domain and Δ is the sampling interval) represents the simple and more direct way of overcoming the unstable sampling zero problem [1,14]. The introduction of delta operator has been given in [1,11,14-17], it has significant advantages in digital control and estimation compared with the simpler shift form and offers the same flexibility as does the shift operator q in the description of discrete models, and gives a better correspondence between continuous and discrete time descriptions.

In this paper, we consider the robust stability problem for continuous time direct model reference adaptive control for non-minimum phase plants in presence of unmodeled dynamics. We apply an exponential input output data filtering to handle the problem of non minimum phase and to reduce the effects of the high frequency modes of the unmodeled dynamics. We use the delta operator to approximate the derivative operator ($D = d/dt$). Thus we transform the continuous time problem into the discrete time one. The proposed data filtering with the application of the Delta operator to the original system model permits to define a new system model which is minimum phased. The stability in the sense that the input and output of the system are bounded for all time is established. Note that the proposed scheme to handle the non-minimum phase is more simple and direct that multirate sampling [12]. To prove the effectiveness and robustness of this new strategy, we will apply it to simulated and real systems.

This paper is organised as follows: In section II, the description of the system is given and the input output data filtering is applied. This data filtering with the application of the operator Delta allows to define a new system model which is minimum phased. In section III, The robustness of the proposed scheme against unmodeled dynamics and noise is established. In section VI, a number of simulation examples and a real application are considered to illustrate the effectiveness of the theoretical results.

II The data filtering

II-1 Problem statement

We consider a continuous time-invariant system input-output pair U(t), Y(t) which are related by:

$$A'(D)Y(t) = B'(D)U(t-d) + \xi(t) \qquad (1)$$

where the $A'(D)$, $B'(D)$ are polynomials in the differential operator $D = (d/dt)$, of order n, m respectively, and d is the time delay. This model has been considered in [13]

DETECTION IN THE DIAGNOSTICS PROBLEMS

A. Naumov, L. Fahrmeir and M. Daumer
Institute for Medical Statistics and Epidemiology,
Ismaninger Str. ,22 ,81675 , Munich ,Germany

Abstracts. This paper is a continuation of investigations of authors in the area of elaborating and modelling new effective methods for detection of change-points and diagnostics. In this paper we propose methods for estimation of diagnoses on the basis of the ideas of opinions functions and detection of characteristical elements by a moving window algorithm and a fixed coordinate system approach.

1.The problem.

Let

$S_i(t)=f_i(t)+\varepsilon_i(t), \quad i=1,2,..., \ n_c,$

$Si : R \to R$, are signals of n_c channels , $f_i(t)$ are nonlinear functions

$\varepsilon i\,(t), \varepsilon i : R \to R$ are noise signals . We have the set D of diagnoses:

$D=D_d \mathrm{Y} D_a$.

Where $D_d=\{D_{d,i}\}, \quad i=1,2,.., \ n_d,$ is the set of object diagnoses and D_a =
set of apparatus (measure devices, channels and so on) diagnoses. We must
signals *Si(t)* . In this paper we consider the methods with small time for tha
the corresponding algorithms in real-time systems. These problems arise in
economics , physics and so on (see Bar [1], Basseville [2], Chan [3], Daume

2. The formal description of the diagnostical problem.

Let the sets of (corresponding to sets D_d and D_a) characteristical element

$$H^{e,d} = \left\{\eta^{e,d}_{i,j}(t)\right\}, \ i = 1,2,..., n_c \ , \ j = 1,2,..., n_i^d \ ;$$

$$H^{e,a} = \left\{\eta^{e,a}_{i,j}(t)\right\}, \ i = 1,2,..., n_c \ , \ j = 1,2,..., n_i^a \ ; \ H^e = H^{e,d} \ \mathrm{Y}$$

In practice may be useful, for example, so basis element of

$\Delta\eta=\{\Delta\eta^e_{ij}(t)\}=\{\Delta\eta^{e,a}_{ij}(t)\}\mathrm{Y}\{\Delta\eta^{e,d}_{ij}(t)\}$ and $\Delta\theta=\{\Delta\theta\eta^e_{ij}\}$

domains):

$$\eta^e_{i,1}(t)=\begin{cases}\mu_0+\alpha_1 t, t\le 0\\ \mu_0+\alpha_2 t, t>0\end{cases}; \ \Delta\eta^e_{i,1}(t)=\{\ \eta^e_{i,1}(t) \mid \alpha 11 \le \alpha 1 \le \alpha 12 , \alpha 21$$

$$\Delta\theta\eta^e_{i,1}=\{(\alpha 1,\alpha 2) \mid \alpha 11 \le \alpha 1 \le \alpha 12 , \alpha 21 \le \alpha 2 \le \alpha 22 \ \} .$$

Here $\mu_0 , \alpha 1 , \alpha 2 , \alpha 11 , \alpha 12 , \alpha 21 , \alpha 22$ are

$\Delta\eta=\{\Delta\eta^e_{ij}(t)\}=\{\Delta\eta^{e,a}_{ij}(t)\}\mathrm{Y}\{\Delta\eta^{e,d}_{ij}(t)\}$ is the set of the perm

$\Delta\theta=\{\Delta\theta\eta^e_{ij}\}=\{\Delta\theta\eta^{e,a}_{ij}\}\mathrm{Y}\{\Delta\theta\eta^{e,d}_{ij}\}$ is the set of the

$\theta\eta^e_{ij}$ of the elements $\eta^e_{i,j}(t)$.

AVERAGING OF VISCOELASTIC AND SHRINKAGE PROPERTIES FOR VISCOELASTIC AGING COMPOSITES

J. Orlik[1] and S.E. Mikhailov[2]

[1] *Institute for industrial and economical mathematics (ITWM)*
Erwin-Schroedinger-Str., 67663 Kaiserslautern, Germany (orlik@itwm.uni-kl.de)
[2] *Wessex Institute of Technology*
Ashurst Lodge, Ashurst, Southampton, SO40 7AA, UK (mik@wessex.ac.uk)

Abstract

A multi-phase composite is considered in this paper. The composite component materials are anisotropic linear viscoelastic and aging (described by the non-convolution Volterra integral operators) and are subjected to isotropic shrinkage and mechanical loads including interface jumps of displacements and tractions. The paper describes some theoretical results about solvability and uniqueness of solution to this problem, asymptotic homogenisation, and 2-scale convergence in appropriate function spaces.

Introduction

Assumption 1 (on the geometry) *The non-homogeneous solid $\Omega \subset \mathbb{R}^n$ is composed of s isotropic or anisotropic viscoelastic materials Ω_l, $l = 1, ..., s$, where Ω_l are generally non-connected disjoint Lipschitz domains, and $\partial\Omega_l$ denotes the boundary of Ω_l. We denote by $\Sigma_{lk} = \partial\Omega_l \cap \partial\Omega_k$ the interfaces between the domains Ω_l and Ω_k. Evidently, $\Sigma_{lk} = \Sigma_{kl}$. If Ω_l and Ω_k have no common boundary, then $\Sigma_{kl} = \emptyset$. The net interface is $\Sigma = \cup_{l=1}^{s} \cup_{k=l+1}^{s} \Sigma_{lk}$ and $\Omega = \cup_{l=1}^{s} \Omega_l \cup \Sigma$, with a Lipschitz external boundary $\partial\Omega = \cup_{l=1}^{s} \partial\Omega_l \backslash \Sigma$. Furthermore, each $\partial\Omega_l \setminus \partial\Omega$ has a positive Lebesgue measure on $\partial\Omega_l$. Let $\partial_u\Omega \subset \partial\Omega$ be a subset of the external boundary and $\partial_u\Omega_l = \partial\Omega_l \cap \partial_u\Omega$. Let $\partial_\sigma\Omega = \partial\Omega \setminus \partial_u\Omega$. Suppose the set of points, that belong to boundaries $\partial\Omega_l$ of more than two different subdomains, or two subdomains and the part of boundary $\partial_u\Omega$, has zero measure on each $\partial\Omega_l$.*

Let the suscripts $i, j, h, k = 1, 2, ..., n$, and summation from 1 to n over repeating subscripts is assumed hereafter. For a solid Ω, we consider the equilibrium equations:

$$\frac{\partial}{\partial x_h}\Big(\Big[\underline{a}_{ihjk}(x)\frac{\partial u_j(x,\cdot)}{\partial x_k}\Big](t) - f_{ih}(x,t)\Big) = f_{i0}(x,t), \quad x \in \Omega \setminus \Sigma \tag{1}$$

with boundary conditions:

$$u_i(x,t) = \chi_i(x,t), \qquad x \in \partial_u\Omega, \tag{2}$$

$$\Big(\Big[\underline{a}_{ihjk}(x)\frac{\partial u_j(x,\cdot)}{\partial x_k}\Big](t) - f_{ih}(x,t)\Big)n_h(x) = \omega_i(x,t), \quad x \in \partial_\sigma\Omega, \tag{3}$$

and transmission conditions:

$$u_i(x,t)\mid_{\Sigma^+} - u_i(x,t)\mid_{\Sigma^-} = \chi_i(x,t), \ x \in \Sigma, \tag{4}$$

$$\begin{aligned}&\left(\left[\underline{a}_{ihjk}(x)\frac{\partial u_j(x,\cdot)}{\partial x_k}\right](t) - f_{ih}(x,t)\right)n_h(x)\Big|_{\Sigma^+} \\ &\quad + \left(\left[\underline{a}_{ihjk}(x)\frac{\partial u_j(x,\cdot)}{\partial x_k}\right](t) - f_{ih}(x,t)\right)n_h(x)\Big|_{\Sigma^-} = \omega_i(x,t), \ x \in \Sigma,\end{aligned} \tag{5}$$

holding for any $t \in [0, T]$. Here $\underline{a}_{ihjk}(x) := a^{\circ}_{ihjk}(x,t) + a_{ihjk}(x)\star$, see e.g. [1]; the out-of-integral term a°_{ihjk} presents the instant elastic coefficients; the Volterra operator $(a_{ihjk}(x) \star e_{jk})(t) := \int_0^t a_{ihjk}(x,t,\tau) \cdot e_{jk}(x,\tau)d\tau$ presents the viscosity with ageing (for isotropic materials $\underline{a}^{hk}_{ij} = \underline{\lambda}\delta_{hi}\delta_{kj} + \underline{\mu}\delta_{ij}\delta_{hk} + \underline{\mu}\delta_{ik}\delta_{hj}$); f_{i0} are components of a vector of external forces; $f_{ih} := \underline{a}_{ihjk}e_{jk}$ are components of the so-called shrinkage stress tensor, where $e_{jk}(x,t)$ is a free shrinkage strain; $\chi_i(x,t) := \{\{\chi_i^{lk}(x,t)\}_{k=l+1}^{s}\}_{l=0}^{s}$, where

ANALITICAL MODEL OF DISCONTINUOUS DRYING PROCESS

L. L. Pezo[1], D. LJ. Debeljkovic[2], D. Voronjec[3]

[1] *Eng. Dept. Holding Institute of General and Physical Chemistry, Studentski trg 12/V, Belgrade, fax: ++381-11-639-624, e-mail: inzenjer@Eunet.yu*

[2] *Dept. of Control Eng., University of Belgrade, Faculty of Mechanical Eng., 27. marta 80, Belgrade, fax: ++381-11-33-70-364, e-mail: ddebeljkovic@alfa.mas.bg.ac.yu*

[3] *Dept. of Thermodynamics., University of Belgrade, Faculty of Mechanical Eng., 27. marta 80, Belgrade.*

Abstract: On the basis of really accepted and critically clarified assumptions mathematical model of discontinuous conductive-convective atmospheric dryer has been developed, in engineering sense, sufficiently correct. The model has a form of partial difference equations system, with coefficients variable in time. This complexity of mathematical model is a result of the fact that nominal values of process variables are changeable in time. Using the simulation of acquired results there has been presented dynamical behavior of this dryer, which is similar to real plant behavior.

Key words: mathematical modeling, dryer, dynamical analysis.

1. Introduction

Nowadays drying system construction is connected with many requirements concerning final product quality and economic parameters, as well as some dynamic characteristics of the system. In order to fulfill all these demands it is necessary to develop a sufficiently correct mathematical model of drying process. In the process of mathematical model forming, it is very important to define the control boundary, by which it is possible to write balance equations of the process.

2. Process description

Zeolite powder conductive-convective atmospheric dryer symbolic-functional scheme is shown on Fig. 1. This dryer is working under atmospheric pressure, and is consisted of aluminum plate, which is put on a heater (electrical heater). Control boundary is shown by doted line. The material is separated, geometrically, in 4 layers (marked *M1, M2, M3, and M4*), as it is shown on Fig. 2. Layers are equal in height. Relevant process variables are medium temperatures of material for all four layers (θ_{DM1}, θ_{DM2}, θ_{DM3}, θ_{DM4}), as well as medium material humidity in all four layers (x_{M1}, x_{M2}, x_{M3}, x_{M4}), as it is shown on Fig. 2. Space above free surface is separated in two layers (*A1, A2*), in which the medium temperature (θ_{DA1}, θ_{DA2}) and medium moisture content is measured (x_{A1}, x_{A1}). Heater is marked by P, and heater temperature is the control variable in drying process. Disturbance variable is environment temperature(θ_E), marked with O.

Wet material (zeolite powder) is carried into plate, before the drying process. Plate is made of aluminum, for better heat-transfer coefficient. Its dimensions are 200 mm x 80-mm. Wall thickness is 1 mm. Plate is placed on the heater, and there is no mixing included, so there is no homogenization during the process. Heat and mass transfer are taking place between the layers (temperature and humidity field is not homogenizing), and that is approved by experimental results. Plate is not insulated. It is necessary to dry 864 g of wet material, whose water content is aprox. 30% in the beginning, and approx. at the end of process. Material is heated from 20 ^{o}C to 160 ^{o}C. According to drying curve, which is obtained by experimental results, drying process is about nine hours long.

3. Experimental measurement

Four measurements have been carried out on the plant described in the preceeding chapter. Their aim was to determine the changes in temperature and moisture content in wet material, as well as, in the air above the free surface of material. On the basis of measured values of these variables for the *first measurement*, during which is not performed temperature regime change, temperature change rate, and humidity change rate have been determined for wet material and for air, and also the first differential of temperature on space coordinate η

A ROBUST MRAC FOR MULTIVARIABLE SYSTEMS

A. Radouane and L. Radouane
LESSI, département de physique
faculté des sciences BP 1796
FES ATLAS 30000; Morocco
E-mail: abdelhayr@hotmail.com

Abstract. In this paper, a robust direct model reference adaptive control for linear discrete-time non minimum phase multivariable systems is proposed. Based on a filtering method, the global stability and robustness of closed-loop control system are achieved even when the unmodelled dynamics are present. Furthermore, the nonsingularity problem is considered. Finally, a simulation study is given for illustration.

1. Introduction

The problem of adaptive control of multi-input multi-output (MIMO) linear time invariant systems has attracted the attention of researchers for several years. While some new and sophisticated control algorithms are being developed, already established ones are being extended to the multivariable case one by one. In the extension of the model reference adaptive control to the multivariable case, time delay proved to be the key issue. Thus, in the MIMO case, the delay structure is defined by a polynomial matrix known as the interactor [6]. It was pointed out in [3] that assuming knowledge of the interactor matrix is tantamount to knowing the full system. An alternative model reference adaptive control (MRAC) is proposed in [4, 13], and consists on estimating the interactor matrix. One of the main problems in the design of MRAC for MIMO linear systems are the singularities that may arise in the control law. The parameter modification procedure allows the problem of the nonsingularity in a different perspective to be dealt with [12]. In the point of view robustness, undesirable transient responses and tracking performances have been frequently observed in traditional MRAC problems, especially for multivariable plants with unmodelled dynamics and output disturbances. An important step was taken in [1, 14] where the robustness of MIMO MRAC schemes is studied by using the variable structure design. In [2], an adaptive variable structure scheme for solving robustness problems in MRAC scheme for MIMO systems is presented. However, the prior knowledge of the interactor matrix is used. A MIMO robust adaptive control using the factorization approach were proposed in [15]. However, this method is restricted to class of MIMO systems. In this paper, we deal with a robust MRAC for discrete-time non minimum phase MIMO systems with unknown interactor matrix, the associated adaptive controller must be robust in the sense that for any unmodelled dynamics, the adaptive closed-loop system is BIBO stable. To avoid the cancellation of non-minimum phase zeros, we will introduce in this paper an input-output data filtering. The latter method permits to ensure the robustness against a large class of unmodelled dynamics. Therefore, the formulation of unmodelled dynamics in this paper, leads to an explicit extension of the modified parameter estimation addressed in [12].
The paper is organized as follows: In section 2, we present the problem formulation. Section 3 is devoted to data filtering. In section 4, we present the one step ahead control law. Section 5 is devoted to adaptive control law. In section 6, a numerical example is given for illustration. Finally, section 7 gives the conclusions of this work.

2. Problem formulation

The class of controlled plants we consider can be modeled as in the equation :

$$A(q^{-1})y(t) = B(q^{-1})u(t) + \psi(t) \tag{1}$$

where $y(t)$, $u(t) \in \Re^r$ are the output and input vectors. $\psi(t) \in \Re^r$ it represents the unmodelled dynamics.

$A(q^{-1})$, $B(q^{-1})$ are polynomial matrices in the unit delay operator q^{-1} and are defined as:

$$A(q^{-1}) = I + A_1 q^{-1} + \ldots + A_n q^{-n} \tag{2}$$

$$B(q^{-I}) = B_1 q^{-1} + \ldots + B_n q^{-n} \tag{3}$$

We make the following assumptions:
A.1 The transfer function $N(q)$ between $y(t)$ and $u(t)$ is strictly propre and has full rank

List of Authors