



ASIM Workshop Wismar -

Modellierung und Simulation in Automotive und Prozessautomation

Mai 8-9, 2003

Wismar

Zusammenfassung der Vorträge

ARGESIM Report no. 25
ASIM - Mitteilungen AMB 85



ARGESIM Report



ARGESIM Report

ISBN 3-901608-25-7





ARGESIM Report no. 25

**C. Deatcu (Hrsg.)
S. Pawletta (Hrsg.)
T. Pawletta (Hrsg.)**

ASIM – Mitteilung AMB 85

**ASIM Workshop Wismar -
Modellierung und Simulation
in Automotive und Prozessautomation**

**Mai 8-9, 2003
Wismar**

organisiert durch:



**Hochschule Wismar
Universität Rostock
IAV GmbH Gifhorn**

Zusammenfassung der Vorträge

**ARGESIM - Verlag, Vienna, 2003
ISBN 3-901608-25-7**

ARGESIM Reports

Published by **ARGESIM** and **ASIM**, Arbeitsgemeinschaft Simulation,
Fachausschuss 4.5 der GI

Series Editor:

Felix Breiteneker (ARGESIM / ASIM)
Div. Simulation, Vienna University of Technology
Wiedner Hauptstrasse 8 - 10, A - 1040 Vienna
Tel: +43-1-58801-11452, Fax: +43-1-58801-11499
Email: Felix.Breiteneker@tuwien.ac.at

ARGESIM Report no. 25 **ASIM – Mitteilung AMB 85**

Titel: ASIM Workshop Wismar –
Modellierung und Simulation in Automotive und Prozessautomation

Herausgeber: C. Deatcu
S. Pawletta
T. Pawletta

Email: c.deatcu@mb.hs-wismar.de

ISBN 3-901608-25-7

Das Werk ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere die der Übersetzung, des Nachdrucks, der Entnahme von Abbildungen, der Funksendung, der Wiedergabe auf photomechanischem oder ähnlichem Weg und der Speicherung in Datenverarbeitungsanlagen bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten.

© by ARGESIM / ASIM, Wien, 2003 – Hochschule Wismar

ARGE Simualtion News (ARGESIM)
c/o F. Breiteneker, Div. Simulation, Vienna Univ. of Technology
Wiedner Hauptstrasse 8-10, A-1040 Vienna, Austria
Tel.: +43-1-58801-11452, Fax: +43-1-58801-42098
Email: info@argesim.org; WWW: <http://www.argesim.org>

Druck:

Copy Shop Gluth, Wismar

Vorwort der Herausgeber

Die Lösung komplexer Probleme mittels Modellbildung und Simulation ist eine etablierte Vorgehensweise und wird in vielen Bereichen der Natur- und Ingenieurwissenschaften bis hin zur Medizin und den Wirtschafts- sowie Sozialwissenschaften angewendet. Neben der mathematischen und softwaretechnischen Weiterentwicklung ist für Ingenieure immer die Anwendung auf neue Problemstellungen interessant und oft ergibt diese entscheidende Impulse für neue methodische Ansätze. Da Anwendungsprobleme am besten in Zusammenarbeit mit Anwendern diskutiert werden können, ist ein wesentliches Ziel des Workshops der Erfahrungsaustausch zwischen den Simulationsspezialisten aus dem Forschungs- sowie Hochschulbereich und den Simulationsanwendern aus der Industrie. Mit Blick auf diese Zielstellung freuen sich die Veranstalter, dass bei den Beiträgen ein ausgewogenes Verhältnis zwischen Hochschul- und Industriethemen besteht, die in mehreren Fällen in Kooperation bearbeitet wurden.

Gemäß dem Workshop-Thema kommen die meisten Beiträge aus dem Bereich der Automobilentwicklung. Daneben gibt es einzelne Beiträge zur simulationsgestützten Prozessautomation und -planung bis hin zu einer Anwendung aus der Medizin beziehungsweise dem High-Performance-Computing. Auch wenn ein Workshop im Gegensatz zu einer Tagung thematisch begrenzter ist, sind die Veranstalter der Meinung, dass ein „kurzer Blick über den Tellerrand“ für alle Teilnehmer interessant und auch in Bezug auf das eigene Gebiet lehrreich sein kann.

Beim Tagungsband gibt es einen kleinen Wehmuts tropfen. Nicht alle Beiträge wurden für eine umfassende schriftliche Publikation freigegeben. Jeder Workshop-Beitrag ist aber mindestens durch eine Zusammenfassung im Tagungsband präsent. Die Veranstalter bedanken sich bei der IAV GmbH Berlin/Gifhorn für die Unterstützung beim Druck des Tagungsbandes und der gesamten Workshopdurchführung. Weiterhin sei Herrn Prof. Breitenacker für seine Unterstützung von Seiten der ASIM gedankt.

Wismar, den 05.05.2003

Christina Deatcu, Sven Pawletta und Thorsten Pawletta

Inhalt

B.P.Lampe

Konvertierung von Modellen für lineare Mehrgrößen-Systeme.....1

S. X Ding, H.-G Schulz

*Modellierung der Fahrquerdynamik und ihre Anwendung
zur Überwachung der Querdynamik von Fahrzeugen.....7*

A. Heim, G. Sauerbier, T. Pawletta

Modellierung und Simulation des dynamischen Verhaltens
von Gas-Feder-Dämpfer-Einheiten.....9

F. Richert, J. Rückert, A. Schloßer

Dymola und Matlab im praktischen Einsatz:
Ein Vergleich anhand der Modellierung eines Dieselmotors.....17

P. Giesemann

*Entwicklung eines steuergerätaefähigen Modells
für einen Drei-Wege-Katalysator.....27*

S. Streck, J. Smirek, V. Fäßler

Modellbildungsstrategien und Berechnungsmethodik
in der Abgasanlagenentwicklung.....29

M. Münzenmay, F. Hirsch, H. Kull

Verwendung einer strömungsmechanischen Motorsimulation als
'virtueller Prüfstand' zur Untersuchung und Bewertung von Regelkonzepten
am Beispiel eines PI-Gleichstellungsreglers für die Frischluftmassenströme
in einem Biturbomotor.....41

E. Klemkow, M. F. Majohr

Materialflusssimulation in der Flugzeugstrukturmontage.....49

M. Kremp, T. Pawletta, S. Pawletta, G. Colquhoun

Bewertung verschiedener Strategien zur simulationsbasierten Steuerung
von Materialflusssystemen.....55

O. Simanski, R. Kähler, B.P. Lampe, B. Pohl, R. Hofmockel

Modellbasierte Regelung in der Anästhesie.....63

P. Roßmanek, C. Wahmkow

*Das intelligente Kraftfahrzeug - mehr Sicherheit
durch eine komplexe Signalverarbeitung mittels Neuro-Fuzzy-Steuerung.....71*

H. Voelker, A. Schluß

Simulation leitungsgebundener elektromagnetischer Phänomene
im KFZ-Bordnetz.....73

J. Nocke

Simulation des motorischen Innenprozesses.....81

R. Fink, S. Pawletta

Verkürzung von Entwurfszeiten durch Matlab-basiertes HPC.....83

K. Komar

Modellbildung und Simulation für die Entwicklung von Motorsteuerungen.....85

(kursiv gedruckte Vorträge sind nur mit kurzem Abstract im Tagungsband enthalten)

Konvertierung von Modellen für lineare Mehrgrößen-Systeme

B.P. Lampe, Universität Rostock *

Zusammenfassung

Der Übergang von einer Beschreibungsform für MIMO Systeme zu einer anderen Beschreibungsform ist im allgemeinen schlecht konditioniert, wodurch inkorrekte Ergebnisse entstehen können. Es wird gezeigt, dass sich durch kleine Abweichungen sogar die Systemordnung verändern kann, was fehlerhafte Reglerentwürfe mit nicht abschätzbaren Auswirkungen auf das Regelsystem zur Folge hat. Die Ursachen für dieses Phänomen sowie Möglichkeiten zu dessen Vermeidung werden angegeben.

1 Einleitung

Viele der in Verbrennungsmotoren ablaufenden Prozesse besitzen mehrere Ausgangsgrößen, die über mehrere Eingangsgrößen beeinflusst werden können. Sie werden als MIMO (Multiple Input Multiple Output) Prozesse bezeichnet. Als Beispiel sei die kombinierte AGR (Abgas-Rückführ-Rate-Regelung) genannt, die mit oder ohne Turbolader sowohl für Benzin als auch für Dieselmotoren von Bedeutung ist.

Bei der Regelung von MIMO Prozessen geht man oft so vor, dass man die hauptsächlichen Wirkungen als isolierte Strecken betrachtet und dafür Regler entwirft. Man hofft dann, dass die Wechselwirkungen wie Störungen von den isolierten Regelkreisen ausgeglichen werden. Bei solchen dezentralen Regelungen müssen die Verkopplungen nur grob bekannt sein [Lun97]. Andererseits ist klar, dass man in den Fällen, wo die Verkopplungen bekannt sind, durch den vereinfachten Ansatz Freiheitsgrade verschenkt, die im Sinne einer besseren Regelgüte genutzt werden können. Das trifft insbesondere dann zu, wenn starke Wechselwirkungen bestehen [Mac89].

Im Falle starker Verkopplungen wird auch der Regler als Mehrgrößen-System angesetzt. Hierdurch können die Wechselwirkungen gleichzeitig mit den Hauptwirkungen ausgeglichen werden. Ausgangspunkt zum Entwurf solcher Regler sind verschiedene Formen von mathematischen Modellen. Am weitesten verbreitet sind lineare Modelle, die im Zustandsraum oder als Eingangs-/Ausgangsmodelle im Frequenzbereich beschrieben werden. Dabei werden in den einzelnen Schritten von der Prozessidentifikation bis zur Implementierung des Mehrgrößen-Reglers unterschiedliche Modelle benutzt [ZDG96].

*Institut für Automatisierungstechnik, D-18051 Rostock, Germany, bernhard.lampe@etechnik.uni-rostock.de

Weitgehend unbekannt ist aber, dass der Übergang von einer Beschreibungsform zu einer anderen Beschreibungsform insofern inkorrekt sein kann, dass kleine Abweichungen in einer Beschreibungsform zu großen Abweichungen in einer anderen Beschreibungsform führen können. Es wird gezeigt, dass sich durch kleine Abweichungen sogar die Systemordnung verändern kann, was fehlerhafte Reglerentwürfe mit nicht abschätzbaren Auswirkungen auf das Regelsystem zur Folge hat. Die Ursachen für dieses Phänomen sowie Möglichkeiten zu dessen Vermeidung werden angegeben.

2 Beschreibungsformen für MIMO LTI Systeme

Im folgenden werden einige der gebräuchlichsten Beschreibungsformen für MIMO LTI Systeme genannt.

2.1 Zustandsraum-Darstellung

Auf der Basis von theoretischer Analyse lassen sich Prozessmodelle der Form

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \quad \mathbf{y} = \mathbf{C}\mathbf{x} \quad (1)$$

aufstellen mit dem Zustand $\mathbf{x} \in \mathbb{R}^p$, dem Eingang $\mathbf{u} \in \mathbb{R}^m$ und dem Ausgang $\mathbf{y} \in \mathbb{R}^n$, in denen $\mathbf{A} \in \mathbb{R}^{p \times p}$, $\mathbf{B} \in \mathbb{R}^{p \times m}$ und $\mathbf{C} \in \mathbb{R}^{n \times p}$ konstante Matrizen sind. Für den Entwurf von Reglern ist wichtig, ob der Prozess steuerbar und beobachtbar ist. Bei gleichem Ein-/Ausgangsverhalten kann ein System durch verschiedene Gleichungen der Form (1) dargestellt werden. Eine Darstellung mit minimalem p ist immer steuerbar und beobachtbar und wird minimale Realisierung genannt. Sie wird im allgemeinen bei im Zustandsraum arbeitenden Regler-Entwurfsverfahren vorausgesetzt.

2.2 Ein-/Ausgangs-Darstellungen

Nach Laplace-Transformation und Eliminierung des Zustands $\mathbf{x}$ gewinnt man aus (1) die Ein-/Ausgangsbeziehung

$$\mathbf{y} = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}\mathbf{u}. \quad (2)$$

Indem man alle Brüche auf den Hauptnenner schreibt, gelangt man für die Übertragungsmatrix $\mathbf{G}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}$ zu

$$\mathbf{G}(s) = \frac{\mathbf{L}(s)}{d(s)} \quad (3)$$

mit der Polynommatrix $\mathbf{L}(s) \in \mathbb{R}^{n \times m}[s]$ und dem skalaren Polynom $d(s)$, wobei $\mathbf{M}(s_i) \neq \mathbf{O}$ für alle Polstellen s_i , d.h. $d(s_i) = 0$ gilt. Sie wird als Standard-Form der Übertragungsmatrix $\mathbf{G}(s)$ bezeichnet. Eine Übertragungsmatrix der Form (3) lässt sich auch durch experimentelle Prozessidentifikation gewinnen, wenn man jede Teilstrecke einzeln identifiziert und auf den Hauptnenner bringt. Anschließend müssen nahe beieinander liegende Pole gleichgesetzt und die Zähler korrigiert werden. Für eine Reihe von Entwurfsverfahren wird das Prozessmodell als linke Matrix Fraction Description (LMFD)

$$\mathbf{G}(s) = \mathbf{M}^{-1}(s)\mathbf{N}(s) \quad (4)$$

mit Polynommatrizen $\mathbf{M}(s) \in \mathbb{R}^{n \times n}(s)$ und $\mathbf{N}(s) \in \mathbb{R}^{m \times n}(s)$ benötigt. Im allgemeinen muss die LMFD (4) irreduzibel, d.h. der Grad von $\det \mathbf{M}(s)$ minimal sein.

3 Realisierungen und Übertragungsmatrizen

Eine zu einer Übertragungsmatrix (3) gehörende Zustandsdarstellung (1) wird Realisierung genannt und durch das Tripel $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ bezeichnet.

3.1 Einfache Realisierungen

Die Matrix $\mathbf{A} \in \mathbb{R}^{p \times p}$ heißt zyklisch, wenn

$$\text{rank}(s_i \mathbf{I} - \mathbf{A}) = p - 1 \quad (5)$$

für alle Eigenwerte s_i von $\mathbf{A}$ gilt [Kai80]. Die Realisierung $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ heißt einfach, wenn die Bedingungen

1. $(\mathbf{A}, \mathbf{B})$ vollständig steuerbar
2. $(\mathbf{A}, \mathbf{C})$ vollständig beobachtbar und
3. $\mathbf{A}$ zyklisch

erfüllt sind. Einfache Realisierungen sind strukturell stabil in dem Sinne, dass sie bei kleinen Änderungen in den Elementen von $\mathbf{A}, \mathbf{B}, \mathbf{C}$ einfach bleiben und beliebige Realisierungen durch zufällige Änderungen in diesen Parametern fast sicher einfach werden [RL00].

3.2 Normale Übertragungsmatrizen

Eine Übertragungsmatrix in der Standard-Form (3) heißt normal, wenn jede Unterdeterminante (Minor) 2. Ordnung von $\mathbf{L}(s)$ durch $d(s)$ teilbar ist. Normale Übertragungsmatrizen sind strukturell instabil in dem Sinne, dass kleine zufällige Änderungen in $\mathbf{L}(s)$ fast sicher eine nicht normale Übertragungsmatrix erzeugen. Eine normale Übertragungsmatrix lässt sich in der Gestalt

$$\mathbf{G}(s) = \frac{\mathbf{P}(s)\mathbf{Q}'(s)}{d(s)} + \mathbf{R}(s) \quad (6)$$

mit Polynommatrizen $\mathbf{P}(s) \in \mathbb{R}^{n \times 1}[s]$, $\mathbf{Q}(s) \in \mathbb{R}^{m \times 1}[s]$ und $\mathbf{R}(s) \in \mathbb{R}^{n \times m}[s]$ schreiben. Durch Reduktion der Freiheitsgrade wurde erreicht, dass die Darstellung (6) strukturell stabil ist. Sie wird deshalb S-Darstellung genannt.

3.3 Zusammenhang

In [RL00] wird gezeigt, dass die Übertragungsmatrix $\mathbf{G}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}$ genau dann normal ist, wenn die Realisierung $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ einfach ist.

Der Übergang von der Zustandsdarstellung (1) zur Ein-/Ausgangs-Darstellung (3) oder (4) und zurück erfordert besondere Vorkehrungen, da strukturell stabile in strukturell instabile Darstellungen ineinander überführt werden. Auch bedürfen Operationen

mit normalen Matrizen wegen der dabei unvermeidbaren Ungenauigkeiten einer besonderen Sorgfalt. Zur Überwindung dieser Instabilität kann die S-Darstellung für normale Matrizen verwendet werden, die ihre besondere Struktur bei beliebiger Variation der Koeffizienten bewahrt. Auf Basis der S-Darstellung wird in [RL00] ein allgemeines Verfahren beschrieben, das zu einer gegebenen gebrochen rationalen Matrix eine in ihrer Nähe liegende normale Matrix sucht. Dieses Verfahren wird Normalisierung genannt.

Für jede normale Übertragungsmatrix (3) lässt sich eine LMFD (4) mit $\det \mathbf{M}(s) = d(s)$ finden [RL00].

Die genannten Beziehungen sind algebraischer Natur. Sie gelten deshalb nicht nur für kontinuierliche Systeme (1) und Übertragungsmatrizen in s sondern auch für zeitdiskrete Systeme und Übertragungsmatrizen in z oder $\zeta = z^{-1}$.

4 Beispiele

An mit MATLAB gerechneten Zahlenbeispielen soll demonstriert werden, wie durch kleine Fehler Modelle von überhöhter Ordnung entstehen.

4.1 Rundungsfehler in der Übertragungsmatrix

Gegeben sei die einfache Zustandsdarstellung

$$\dot{\mathbf{x}} = \begin{bmatrix} -1 & 0 \\ 0 & -2 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \mathbf{u}, \quad \mathbf{y} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \mathbf{x} \quad (7)$$

zu der die normale Übertragungsmatrix

$$\mathbf{G}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} = \frac{\begin{bmatrix} s+1 & 0 \\ 0 & s+2 \end{bmatrix}}{(s+1)(s+2)} \quad (8)$$

gehört. Wenn (durch Rundungsfehler) die Übertragungsmatrix etwas verändert wird, etwa zu

$$\tilde{\mathbf{G}}(s) = \frac{\begin{bmatrix} s+1.001 & 0.001 \\ 0 & s+2.001 \end{bmatrix}}{(s+1)(s+2)} \quad (9)$$

dann ist die Matrix $\tilde{\mathbf{G}}(s)$ nicht mehr normal. Zur Übertragungsmatrix $\tilde{\mathbf{G}}(s)$ erzeugt MATLAB die Realisierung $(\tilde{\mathbf{A}}, \tilde{\mathbf{B}}, \tilde{\mathbf{C}})$ mit

$$\tilde{\mathbf{A}} = \begin{bmatrix} -3 & -0.5 & 0 & 0 \\ 4 & 0 & 0 & 0 \\ 0 & 0 & -3 & -0.5 \\ 0 & 0 & 4 & 0 \end{bmatrix} \quad (10)$$

$$\tilde{\mathbf{B}} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad \tilde{\mathbf{C}} = \begin{bmatrix} 1 & 0.2502 & 0 & 0.003 \\ 0 & 0 & 1 & 0.5002 \end{bmatrix}$$

und diese ist eine minimale Realisierung, da sie durch den Befehl `minreal` mit Toleranz `tol=0.001` nicht verändert wird. Reglerentwürfe auf Basis des Modells (10) werden in der Regel unbrauchbare Ergebnisse liefern.

4.2 Prozessidentifikation im Frequenzbereich

Die Elemente $G_{ik}(s)$ der Übertragungsmatrix $\mathbf{G}(s)$ können einzeln bestimmt werden, indem man nacheinander die Eingänge u_k verändert und die zugehörigen Änderungen an den Ausgängen y_i , $i = 1, \dots, n$ bestimmt. Danach würde man üblicherweise so vorgehen, dass man nahe beieinander liegende Polstellen der Übertragungsfunktionen $G_{ik}(s)$ zwangsweise gleich setzt und mit diesem Ansatz die Rechnungen wiederholt. Man gelangt so zur Standard-Form (3). Während man die Bindungen zwischen den Polstellen auf diese Weise berücksichtigen kann, bleiben die in der Polynommatrix $\mathbf{M}(s)$ vorhandenen Bindungen unberücksichtigt. Das folgende Beispiel zeigt, dass dadurch ungeeignete Prozessmodelle generiert werden.

Das tatsächliche System werde durch die einfache Realisierung $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ mit

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 1 & 0 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix} \quad (11)$$

beschrieben, zu der die Übertragungsmatrix

$$\mathbf{G}(s) = \frac{\begin{bmatrix} 2s^2 - 7s + 5 & 0 \\ s^2 - 4s + 3 & s^2 - 5s + 6 \end{bmatrix}}{s^3 - 6s^2 + 11s - 6} \quad (12)$$

gehört, die selbstverständlich normal ist. Nach dem oben beschriebenen Verfahren wurde unter stochastischen Störungen und bei Kenntnis von $G_{12}(s) = 0$ in [AO95] die Übertragungsmatrix

$$\tilde{\mathbf{G}}(s) = \frac{\begin{bmatrix} 1.9992s^2 - 5.4242s + 3.4250 & 0 \\ 0.9991s^2 - 3.1858s + 1.6954 & s^2 - 4.2053s + 3.9990 \end{bmatrix}}{s^3 - 5.2053s^2 + 8.2042s - 3.9980} \quad (13)$$

geschätzt. Diese Matrix ist nicht normal. Zu (13) liefert MATLAB bei Toleranz `tol=0.01` die minimale Realisierung $(\tilde{\mathbf{A}}, \tilde{\mathbf{B}}, \tilde{\mathbf{C}})$ mit

$$\tilde{\mathbf{A}} = \begin{bmatrix} 5.205 & -1.026 & 0.2499 & 0 & 0 & 0 \\ 8 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5.205 & -1.026 & 0.2499 \\ 0 & 0 & 0 & 8 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 \end{bmatrix} \quad (14)$$

$$\tilde{\mathbf{B}} = \begin{bmatrix} 2 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 2 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad \tilde{\mathbf{C}} = \begin{bmatrix} 0.9996 & -0.339 & 0.107 & 0 & 0 & 0 \\ 0.4995 & -0.1991 & 0.05298 & 0.5 & -0.2628 & 0.125 \end{bmatrix}.$$

In [RFL97] wurde gezeigt, dass auch die MFD von (13) einen zu hohen Grad besitzt.

5 Zusammenfassung

Im Artikel wird auf die Gefahren hingewiesen, die bei der Konvertierung von verschiedenen Beschreibungsformen für MIMO LTI Systeme entstehen können. Die Ursachen dafür sind algebraischer Natur und Probleme treten zum Beispiel als Folge von Rundungs- oder Schätzfehlern auf. Bekannte Entwicklungswerkzeuge wie MATLAB treffen weder Vor-sorge noch weisen sie auf diese Gefahren hin. Da die Ordnung der Systeme durch die Konvertierung vergrößert werden kann, sind strukturelle Einschränkungen oder anschließende Korrekturen unerlässlich.

Literatur

- [AO95] A.G. Alexandrov and Y.F. Orlov. Finite frequency identification of multivariable objects. In *Proc. 2nd Russian-Swedish Control Conf. (RSCC'95)*, pages 66–69, Saint Petersburg, Russia, 1995.
- [Kai80] T. Kailath. *Linear Systems*. Prentice Hall, Englewood Cliffs, NJ, 1980.
- [Lun97] J. Lunze. *Regelungstechnik 2 - Mehrgrößensysteme, Digitale Regelung*. Springer-Verlag, Berlin, Heidelberg, ..., 1997.
- [Mac89] J.M. Maciejowski. *Multivariable feedback design*. Addison-Wesley, Wokingham, England a.o., 1989.
- [RFL97] E.N. Rosenwasser, P.G. Fedorov, and B.P. Lampe. Construction of MFD-representation of real rational transfer matrices on basis of normalisation procedure. In *Int. Conf. on Computer Methods for Control Systems*, pages 39–42, Szczecin, Poland, December 1997.
- [RL00] E.N. Rosenwasser and B.P. Lampe. *Algebraische Methoden zur Theorie der Mehrgrößen-Abtastsysteme*. Universitätsverlag, Rostock, 2000. ISBN 3-86009-195-6.
- [ZDG96] K. Zhou, J.C. Doyle, and K. Glover. *Robust and optimal control*. Prentice-Hall, Englewood Cliffs, NJ, 1996.

Modellierung der Fahrquerdynamik und ihre Anwendung zur Überwachung der Querdynamik von Fahrzeugen

S. Ding und H.-G. Schulz

**Fachgebiet Automatisierungstechnik und Komplexe Systeme
Universität Duisburg-Essen**

Die stets erhöhten Anforderungen an Fahrsicherheit und Fahrkomfort und der immer verhärtete Wettbewerb in der Automobilindustrie weltweit führen dazu, dass immer mehr neue Funktionskomponenten neben Standardkomponenten wie ABS, ASR oder GMR ins Fahrzeugsystem integriert werden. Die rasante Entwicklung der Mikrosystem-, Mikrocomputer- und Kommunikationstechnik hat diesen Prozess in den letzten Jahren enorm beschleunigt, indem eine Vielzahl von Sensoren, hoch intelligente regelungstechnische Systeme, leistungsfähige Mikroprozessoren und Softwaremodule in Fahrzeugsystemen eingesetzt werden. Eine plausible Regelung setzt voraus, dass der Ist-Zustand der Regelstrecke korrekt gemessen oder nachgebildet wird. Besonders wichtig ist dies bei der Fahrstabilitätsregelung bei extremen Fahrsituationen, wo eine Regelabweichung schon in einer sehr kurzen Zeit ausgeregelt werden muss. Eine On-Line-Überwachung ist daher ein Bestandteil regelungstechnischer Systeme im Fahrzeug.

Es wurde zwecks On-Line-Überwachung der Fahrquerdynamik zunächst ein Modell zur Beschreibung der Querdynamik von Fahrzeugen aufgestellt. Auf Grund der Parameterschwankung im Zusammenhang mit unterschiedlichen Fahrsituationen wurden zur Schätzung der Fahrquerdynamik verschiedene Beobachterkonzepte entwickelt, u.a. robuste Beobachter und adaptive Beobachter. Ferner wurden diese Beobachter zur Entdeckung von Sensorfehlern angewendet. Anhand von Simulation in der Software-Umgebung von MATLAB/SIMULINK/CARSIM wurden diese Überwachungsverfahren erprobt. In diesem Vortrag werden das Querdynamikmodell, die Beobachterkonzepte und einige Simulationsergebnisse vorgestellt.

Modellierung und Simulation des dynamischen Verhaltens von Gas-Feder-Dämpfer-Einheiten (GFD-Einheiten)

A.Heim, G. Sauerbier, ehemals PNP Luftfedersysteme GmbH Crivitz;
T. Pawletta, Hochschule Wismar;

1. Einführung

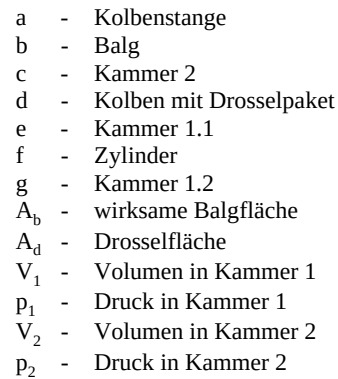
Gas-Feder-Dämpfer-Einheiten federn und dämpfen ausschließlich mit Luft. Durch Zuführen oder Ablassen von Luft kann die Federsteifigkeit dem jeweiligen Lastzustand angepaßt werden. Weiterhin zeigen GFD-Einheiten ein frequenzselektives Dämpfungsverhalten. Dies bedeutet, dass bei entsprechender Auslegung eine bestimmte Frequenz stark bedämpft wird, wogegen die Dämpferkräfte bei hohen Anregungsfrequenzen im Gegensatz zu hydraulischen Dämpfern wieder abnehmen. Diese Eigenschaften erleichtern beim Einsatz in Fahrzeugen Fahrwerksabstimmungen und wirken sich positiv auf das Fahrverhalten aus [1]. Ein Nachteil von GFD-Einheiten gegenüber hydraulischen Dämpfern ist die Notwendigkeit einer zusätzlichen Versorgungseinheit für den Luftbedarf.

Die konstruktive Auslegung von GFD-Einheiten ist aufgrund der komplexen Wechselwirkungen zwischen Gasfeder und Gasdämpfer relativ aufwendig. Um die Anzahl der notwendigen Konstruktionsschritte, vom ersten Prototypen bis hin zur optimal abgestimmten GFD-Einheit zu verringern, ist es sinnvoll, das dynamische Verhalten mittels Computersimulationen zu untersuchen.

Ausgehend vom Funktionsprinzip einer GFD-Einheit wird im Beitrag ein mathematisches Modell zur Beschreibung des dynamischen Verhaltens von GFD-Einheiten aufgestellt. Das Modell enthält physikalische Größen, die nicht als Ausgangsparameter vorliegen und nicht deduktiv ableitbar sind. Für diese wird ein Ansatz zur Identifikation aus Messdaten aufgezeigt. Anschließend werden Einsatzmöglichkeiten und sinnvolle Erweiterungen des Modellansatzes diskutiert sowie die derzeitig entwickelten Programmkomponenten kurz vorgestellt. Aufgrund der Kürze des Beitrages ist es nicht möglich alle Beziehungen ausführlich herzuleiten. Zur detaillierten Beschreibung des mathematischen Modells und der entwickelten Programmkomponenten sowie zur Berechnung von Simulationsszenarien sei auf [4] verwiesen.

2. Funktionsprinzip einer GFD-Einheit

In Abbildung 1 ist der prinzipielle Aufbau einer 3-Raum-GFD-Einheit gemäß [2] dargestellt. Die beiden Kammern 1.1 und 1.2 sind über eine Kolbenstange direkt miteinander verbunden und können deshalb im folgenden vereinfachend als eine Kammer 1 betrachtet werden. Wird die GFD-Einheit zusammengeschoben, verringert sich das Luftvolumen in Kammer 1 und das Volumen in Kammer 2 erhöht sich. Beim Ausfahren kehren sich die Verhältnisse entsprechend um.



3. Modellbildung

F = Kraft an der GED-Ein

- F - Kraft an der GFD-Einheit
- F_R - resultierende Reibungskraft
- p_1 - momentaner Druck in Kammer 1
- p_2 - momentaner Druck in Kammer 2
- p_{aus} - Umgebungsdruck
- A_b - momentan wirksame Balgfläche
- A_d - Drosselfläche
- z - Auslenkung (Anregung)

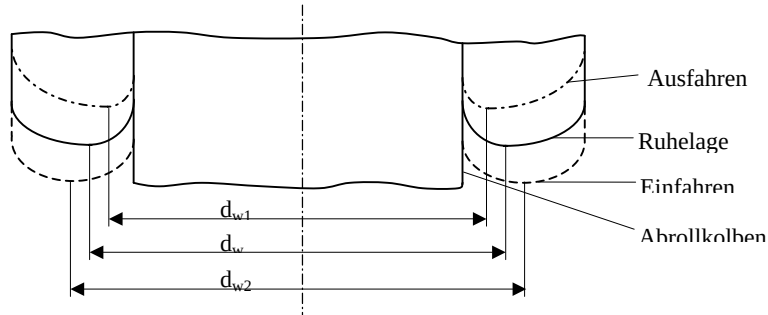


Abbildung 2: Veränderung des wirkenden Balgdurchmessers beim Ein- und Ausfahren

In [4] erfolgte eine ausführliche Untersuchung des Balgverhaltens. Dabei wurde festgestellt, dass dieses durch folgende Näherungsfunktion approximiert werden kann:

$$A_b^{(z)} = a \cdot \left(1 - e^{-\frac{z}{\tau}} \right) + A_{b0}$$

- A_b - momentan wirksame Balgfläche
- A_{b0} - wirksame Balgfläche in Ruhelage
- z - Auslenkung
- a - max. Flächenänderung
- τ - Formparameter

Die Berechnung der sich aufgrund der Erregung ändernden Drücke p_1 und p_2 kann unter Ausnutzung der physikalischen Gesetze der Zustandsänderungen idealer Gase ([5], [6]) erfolgen. Werden diese entsprechend umgeformt und in die Kraftbilanz eingesetzt, erhält man für die an einer GFD-Einheit auftretende Gesamtkraft:¹

$$F^{(t)} = \left\langle p_0 \cdot \left\{ \frac{(m_{10} + m_{\bar{u}}^{(t)}) \cdot R \cdot T_0}{p_0 \cdot [V_{10} + z^{(t)} \cdot (A_d + A_b^{(z)})]} \right\}^{\kappa} - p_{aus} \right\rangle \cdot A_b^{(z)}$$

$$+ \left\langle \left\{ \frac{(m_{10} + m_{\bar{u}}^{(t)}) \cdot R \cdot T_0}{p_0 \cdot [V_{10} + z^{(t)} \cdot (A_d + A_b^{(z)})]} \right\}^{\kappa} - \left\{ \frac{(m_{20} - m_{\bar{u}}^{(t)}) \cdot R \cdot T_0}{p_0 \cdot [V_{20} - z^{(t)} \cdot A_d^{(z)}]} \right\}^{\kappa} \right\rangle \cdot p_0 \cdot A_d - \text{sign}(\dot{z}) \cdot F_R$$

- $m_{10/20}$ - Gasmasse in Kammer 1/2 in Konstruktionslage
- $V_{10/20}$ - Volumen in Kammer 1/2 in Konstruktionslage
- p_0 - Druck in der GFD in Konstruktionslage
- $m_{\bar{u}}$ - über die Drossel geströmte Luftmasse
- R - Gaskonstante
- T_0 - Umgebungstemperatur
- z - Auslenkung (Erregung)
- κ - Adiabaten- oder Polytropenexponent

In dieser Gleichung tritt als unbekannte veränderliche Größe die über die Drossel strömende Gasmasse $m_{\bar{u}}$ auf. Diese ist von der momentanen Druckdifferenz (p_1 - p_2) sowie der konstruktiven Auslegung der Drossel abhängig und kann aufgrund der konstruktiven Komplexität nicht deduktiv abgeleitet werden.

Die über die Drossel strömende Gasmasse kann derzeit nicht direkt gemessen werden. Als Messdaten liegen lediglich Kraftmessungen an GFD-Einheiten bei sinusförmiger Anregung vor [3]. Abbildung 3 zeigt beispielhaft einen an einer realen GFD-Einheit gemessenen Kraftverlauf. Zusätzlich sind in Abbildung 3 die Grenzfälle der Kraftverläufe bei vollständig geöffneter sowie vollständig geschlossener Drossel dargestellt. Die Kraftdifferenzen zwischen dem realen Kraftverlauf und dem bei vollständig geschlossener Drossel stellen ein Maß für die

¹ Die Konstruktionslage ist die Lage der GFD-Einheit im Einbauzustand ohne externe Anregung. In der Konstruktionslage ist eine GFD-Einheit bereits vorgespannt.

zu diesem Zeitpunkt über die Drossel geströmte Gasmasse dar. Von besonderer Bedeutung für die Identifikation sind die beiden Schnittpunkte der drei Kraftverläufe, da abhängig von der Bewegungsrichtung bei diesen Auslenkungen ein Druckgleichgewicht zwischen den Kammern herrscht. Unter Einbeziehung der geometrischen Parameter einer GFD-Einheit können mittels des verbal aufgezeigten Ansatzes die Drosselkennlinien getrennt für die Druck- (Einfahrbewegung) und die Zugstufe (Ausfahrbewegung) ermittelt werden. Eine Drosselkennlinie beschreibt den funktionalen Zusammenhang zwischen der an der Drossel auftretenden Druckdifferenz bei und dem daraus resultierenden Volumenstrom über die Drossel. Aus dem Volumenstrom kann ausgehend vom momentanen Zustand des Gases in den Kammern der GFD-Einheit der Massenstrom $\dot{m}_i$ berechnet werden.

Hinsichtlich der ausführlichen Darstellung des Identifikationsansatzes sei auf [4] verwiesen.

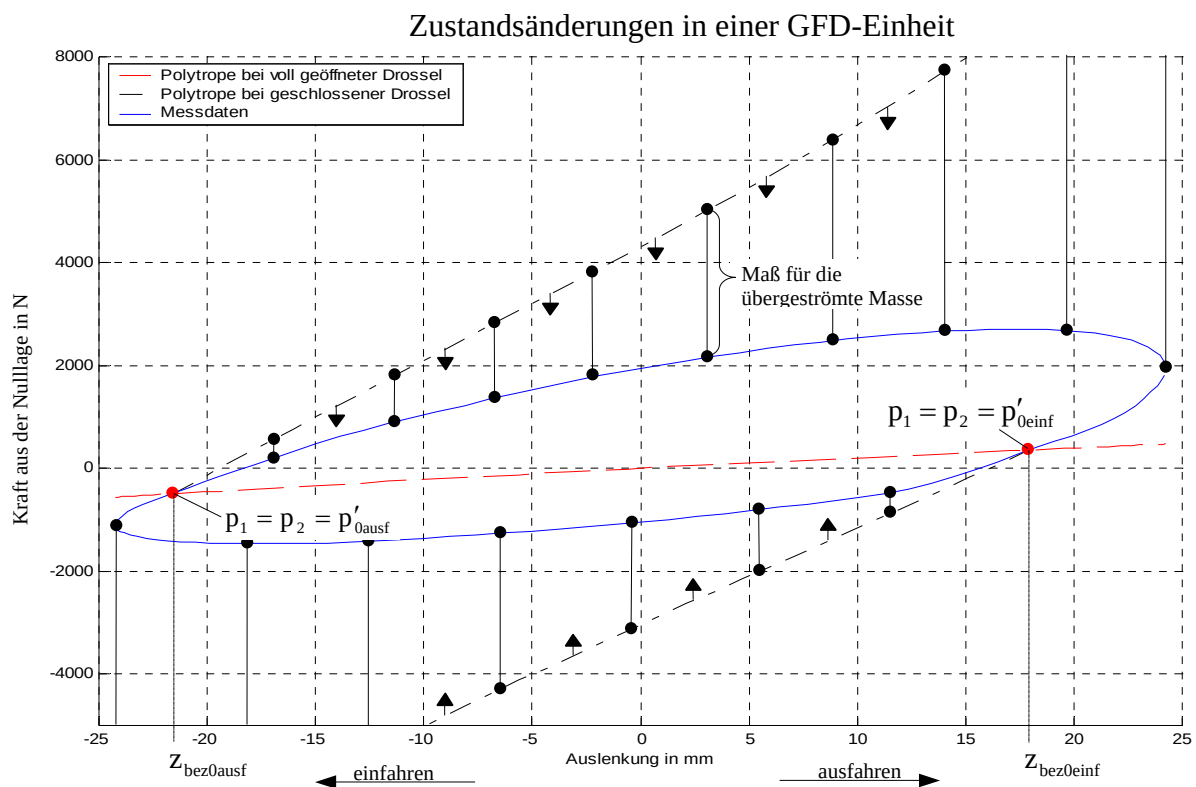


Abbildung 3: Zustandsänderungen in einer GFD-Einheit²

Aus den vorangegangenen Erläuterungen ist zu erkennen, dass der Zustand einer GFD-Einheit durch die Drücke p_1 und p_2 sowie die überströmende Gasmasse $\dot{m}_i$ charakterisiert ist. Aus diesen kann direkt der Kraftverlauf an der GFD-Einheit berechnet werden. .

Um den zeitlichen Verlauf der Zustandsgrößen durch ein numerisches Integrationsverfahren berechnen zu können, müssen die ersten Ableitungen der

² Die Kraft in der Nulllage ist um den Betrag der Kraft in der Konstruktionslage (Vorspannung) korrigiert.

Größen nach der Zeit bekannt sein. Gemäß den in der Kraftbilanz auftretenden Termen für die Gasdrücke p_1 und p_2 ergibt sich für die Dynamikbeschreibung das folgende Differentialgleichungssystem:

$$\begin{aligned} \dot{m}_{\ddot{u}} &= f(\Delta p, v, t) & \text{mit: } v &= \frac{V}{m} \\ \dot{p}_1 &= p_0 \cdot \left(\frac{R \cdot T_0}{p_0} \right)^\kappa \cdot \left\{ \kappa \cdot (m_{10} + m_{\ddot{u}}^{(t)})^{\kappa-1} \cdot \dot{m}_{\ddot{u}} \cdot [V_{10} + z^{(t)} \cdot (A_d + A_b^{(z)})]^{-\kappa} + \right. \\ & \quad \left. (m_{10} + m_{\ddot{u}}^{(t)})^\kappa \cdot -\kappa \cdot [V_{10} + z^{(t)} \cdot (A_d + A_b^{(z)})]^{-\kappa-1} \cdot \dot{z} \cdot (A_d + A_b^{(z)}) \right\} \\ \dot{p}_2 &= p_0 \cdot \left(\frac{R \cdot T_0}{p_0} \right)^\kappa \cdot \left\{ \kappa \cdot (m_{20} - m_{\ddot{u}}^{(t)})^{\kappa-1} \cdot -\dot{m}_{\ddot{u}} \cdot [V_{20} - z^{(t)} \cdot A_d]^{-\kappa} - \right. \\ & \quad \left. (m_{20} + m_{\ddot{u}}^{(t)})^\kappa \cdot \kappa \cdot [V_{20} - z^{(t)} \cdot A_d]^{-\kappa-1} \cdot \dot{z} \cdot A_d \right\} \end{aligned}$$

4. Einsatzmöglichkeiten des Modells

Das aufgestellte Modell ist nur bedingt zur reinen Berechnung des Kraftverlaufes an einer angeregten GFD-Einheit nutzbar, da zur Identifikation der über die Drossel strömenden Gasmasse bereits entsprechende Kraftverläufe vorliegen müssen.

Sehr gut einsetzbar ist das Modell zur Identifikation von Drosselkennlinien und zur Integration in komplexere Modelluntersuchungen, wo das dynamische Verhalten eines Systems mit integrierten GFD-Einheiten zu untersuchen ist. In [4] erfolgte z.B. die Untersuchung des dynamischen Verhaltens eines Viertelfahrzeuges (komplette Radaufhängung) mit integrierter GFD-Einheit.

Ein weiterer Einsatz des Modells ergibt sich daraus, dass bei der Konstruktion von GFD-Einheiten oft ein Eigenschaftsprofil in Form zu realisierender Kraftverläufe bei gewissen Anregungen vorgegeben wird und die dafür notwendigen Konstruktionsparameter der GFD-Einheit zu bestimmen sind. Die diesbezügliche Umformung und Erweiterung des Modells ist der gegenwärtige Entwicklungsschwerpunkt.

5. Programmkomponenten

Das entwickelte Simulationsprogramm besteht aus modular aufgebauten Programmkomponenten, die sowohl einzeln als auch im Verbund benutzt werden können. Jede Programmkomponente verfügt über ein eigenes graphisches Nutzer-Interface, welches die Verbindung mit anderen Programmkomponenten unterstützt. Die nachfolgende Tabelle gibt einen Überblick über die gegenwärtig verfügbaren Programmkomponenten.

Die programmtechnische Umsetzung erfolgte mit dem Programmsystem MATLAB.

Programmkomponente	Funktionen
gfdsim.m	Eingabe aller Parameter einer GFD-Einheit; Einstellung von vordefinierten Drosselkennlinien
auswert.m	Einlesen und Aufbereitung von Kraftverläufen aus Messdaten
balgeinfl.m	Bestimmung von BalghystereseFunktionen
drossel.m	Identifikation von Drosselkennlinien aus Kraftverläufen
kwd.m	Simulation von Kraftverläufen unter Verwendung festgelegter Drosselkennlinien für die Zug- und Druckstufe
zmschw.m	Simulation der Aufbauvergrößerungsfunktion eines Viertelfahrzeuges mit integrierter GFD-Einheit unter Verwendung festgelegter Drosselkennlinien für die Zug- und Druckstufe

6. Zusammenfassung und Ausblick

Es wurde ein mathematisches Modell zur Beschreibung des dynamischen Verhaltens von GFD-Einheiten hergeleitet und darauf aufbauend ein aus mehreren modularen Komponenten bestehendes Programmsystem entwickelt.

Ein Schwachpunkt der gegenwärtigen Modellbildung besteht in der nur experimentell möglichen Bestimmung des Drosselverhaltens. Allerdings gibt es diesbezüglich aufgrund der konstruktiven Komplexität der eingesetzten Drosseln gegenwärtig kaum eine Aussicht auf eine deduktiv ableitbare Beschreibung.

Bei in [4] durchgeführten Simulationsexperimenten zeigte sich, dass die Simulationsergebnisse eine sehr gute Übereinstimmung mit dem auf einem Prüfstand gemessenen dynamischen Verhalten von GFD-Einheiten aufweisen. Weiterhin wurde in [4] am Beispiel der Simulation des dynamischen Verhaltens eines Viertelfahrzeuges mit integrierter GFD-Einheit nachgewiesen, dass der entwickelte Modellansatz und die ihn realisierenden Programmkomponenten sich problemlos in komplexere Aufgabenstellungen einbinden lassen.

Die weitere Arbeit soll sich einerseits auf die Einbindung des Modells in komplexere Problemstellungen, z.B. in ein Nickschwingsystem eines Motorrades, konzentrieren und andererseits auf die Erweiterung und Umformung des Modells in Richtung einer direkten Bestimmung von Konstruktionsparametern von GFD-Einheiten bei vorgeschriebenen Eigenschaftsprofilen.

Literatur

- [1] W. Schwarz, Luftfahrt, In: Motorrad, 18/1990
- [2] Konstruktionsunterlagen von GFD-Einheiten, PNP Luftfedersysteme GmbH, 2000
- [3] Messdaten einer GFD-Einheit an einem VDA-Prüfstand, PNP Luftfedersysteme GmbH, 2000
- [4] A. Heim, Modellierung und Simulation des dynamischen Verhaltens von GFD-Einheiten, Diplomarbeit, Hochschule Wismar, 2000
- [5] W. Keitz, K.-H. Küttner, Dubbel-Taschenbuch für den Maschinenbau, Springer-Verlag Berlin, 1994
- [6] H. D. Baer, Thermodynamik, Springer-Verlag Berlin, 1992
- [7] H. Gold, Über das Dämpfungsverhalten von Kraftfahrzeug-Gasfedern, Dissertation, RWTH Aachen, 1973

Dymola und Matlab im praktischen Einsatz: Ein Vergleich anhand der Modellierung eines Dieselmotors

F. Richert, J. Rückert, A. Schloßer, RWTH Aachen *

Zusammenfassung

Modellbildung und Simulation beschäftigen sich häufig nicht nur mit der reinen Systembeschreibung. Vielmehr basieren viele Probleme, die in der täglichen Arbeit gelöst werden müssen, auf typischen Hürden der Simulationsprogramme und üblichen Anforderungen an ein Modell, wie z.B. modularer und hierarchischer Aufbau, Wiederverwendbarkeit und Erweiterbarkeit. Die objektorientierte Modellierungssprache Modelica bietet einen gegenüber der meist üblichen, signalbasierten Modellierung einen vielversprechenden Ansatz. Dabei konzentriert sich der Modellierer primär auf die Beschreibung der Systeme; die notwendigen Gleichungsmanipulationen und Strukturierung der Signale wird ihm dabei soweit wie möglich vom Simulator abgenommen. Ob dieser Anspruch auch bei komplexeren Modellen zu erfüllen ist, wurde untersucht, indem ein in Matlab/Simulink vorhandenes Modell eines Dieselmotors nach Modelica/Dymola portiert wurde.

1 Einleitung

Die Bedeutung von Modellbildung und Simulation in allen technischen Bereichen stellt heute niemand mehr in Frage. Es gibt kaum einen Bereich, in dem nicht mit Hilfe von Simulationen Zeit und Geld gespart, Risiko minimiert oder messtechnisch unzugängliche Systeme untersucht werden. Die Zahl der dafür existierenden Werkzeuge ist dabei mindestens so groß wie die der unterschiedlichen Einsatzgebiete. Eine große Oberfläche ist dabei mittlerweile ebenso Stand der Technik, wie die Möglichkeit, die Modelle hierarchisch und modular zu strukturieren. So ist unter anderem gewährleistet, dass die Modellstruktur alleine schon eine gewisse Dokumentation darstellt. Dabei orientieren sich die meisten Programme an der aus der Regelungstechnik bekannten Darstellung von Systemen mittels Wirkungsplänen. Diese bieten u.a. den Vorteil, dass sie mathematische Gleichungen unmittelbar grafisch abbilden können. Intern ist allen Simulatoren von detaillierten physikalischen Prozessrechnungen bis hin zu kleinen Programmen „für die kleine Simulation zwischendurch“ wie z.B. *JaLa* [1] jedoch gemeinsam, dass sie ein System von Differential-Algebraischen Gleichungen (bzw. besser bekannt unter dem englischen

*Institut für Regelungstechnik, Steinbachstraße 54, 52072 Aachen

Ausdruck Differential Algebraic Equation, DAE) erzeugen, das in irgendeiner Form numerisch gelöst werden muss. Diese Gleichungen werden dabei aus der gra'schen Struktur erzeugt und orientieren sich auch in ihrem Aufbau daran. Dies führt häufig zu systembedingten Problemen, so können z.B. sogenannte „algebraische Schleifen“ entstehen, deren Lösung in den Programmen selbst im linearen Fall iterativ erfolgt.

Modelica bietet einen anderen Ansatz: Die Modellbildung endet komponentenbasiert, d.h. objektorientiert, statt, die Modellstruktur gleicht dem realen System viel stärker, da physikalische Größen statt gerichteter Signale als Schnittstellen dienen.

Die Grundlagen der Modellierung mit Modelica sind bereits in hinreichend vielen Artikeln beschrieben worden, z.B. [2] und [3] Teil 1-3. daher wird hier nur der Vollständigkeit halber kurz das Modellierungskonzept erläutert.

Um diese neue Modellierungs-Sprache in Verbindung mit dem eigentlichen Interpreter/Simulator und der gra'schen Oberfläche Dymola beurteilen zu können, wurde anhand eines Modells eines Dieselmotors der direkte Vergleich zwischen Modelica/Dymola und Matlab/Simulink gezogen. Dieses Mittelwert-Modell existiert bereits als Umsetzung unter Matlab/Simulink und dient unter anderem zur Reglerentwicklung („Rapid Control Prototyping“, RCP), für Ladedruck- und Abgasrückführaten-Regelung [4], [5]. Neben der reinen Simulation stellt dies zusätzliche Anforderungen nach Werkzeugen zur Auswertung und Visualisierung. Dieses Modell wurde basierend auf identischen Gleichungen unter Modelica/Dymola umgesetzt. Im vorliegenden Artikel wird anhand ausgewählter Beispiele ein Auszug aus dem Vergleich gegeben. Dabei werden besonders alle für den praktischen Einsatz interessanten Aspekte betrachtet, angefangen mit der Bedienbarkeit und dem Aufbau der Modelle über deren Genauigkeit und Stabilität bis hin zu den Möglichkeiten der geforderten Auswertung. Eine Beschreibung der im Modell implementierten Gleichungen würde hier den verfügbaren Rahmen sprengen. Auszüge oder diesem Modell zugrunde liegende Arbeiten finden sich z.B. in [6] und [7].

2 Vorgehensweise

Mit *Modelica* und *Dymola* müssen vorab zwei Begriffe unterschieden werden, die meist nur gemeinsam verwendet werden:

Modelica ist eine reine Modellierungs-Sprache und ist frei verfügbar. Durch *Modelica* wird lediglich die Struktur der Modelle als Quelltext in ASCII-Notation vorgegeben. Werkzeuge für deren Simulation sind nicht Bestandteil von *Modelica*. Die Erstellung eines Modells kann somit vollständig textuell erfolgen.

Dymola ist ein kommerzielles Simulations-Programm, das eine gra'sche Oberfläche zur Erstellung von Modellen in *Modelica*-Syntax bietet. Zwar wird die Arbeit direkt am Code zugelassen und gut unterstützt, ist jedoch nicht zwingend notwendig. Durch intuitives *Drag-and-Drop* werden Modelle aus mitgelieferten Standard-Bibliotheken zusammengesetzt. Die für die Grafik notwendigen Zusatzinformationen werden dabei in den dafür von *Modelica* vorgesehenen *annotations* als Kommentar im Quelltext der Modelle abgelegt.

Weiterhin ist *Dymola* der eigentliche Interpreter der Modelle. Insbesondere die Auswertung und Verarbeitung der Modelle hin zu einer für Rechner nutzbaren Form obliegt

diesem Werkzeug. Die gesamte Simulation und Visualisierung findet in Dymola statt. Beide Bestandteile sind stark verzahnt und lassen sich in der Praxis kaum getrennt behandeln. Modelle können auch rein textbasiert erstellt werden [8], zum Simulieren wird jedoch ein entsprechendes Programm benötigt. Neben Dymola und MathModelica ist den Autoren kein weiteres Programm bekannt, das Modelica-Modelle verarbeiten kann. Die Verknüpfung zwischen Matlab und Simulink ist zwar prinzipiell anders gelagert, dennoch lassen sich für die Anwendung die Begriffspaare MATLAB/SIMULINK und MODELICA/DYMOLA gut gegenüberstellen.

Bei den diesem Artikel zu Grunde liegenden Arbeiten wurden alle Modelle vollständig neu aufgebaut. Es existieren zwar zahlreiche fertige oder halbfertige Bibliotheken für eine Vielzahl von z.B. mechanischen, elektrischen oder thermo-hydraulischen Elementen [9], auf deren Verwendung wurde zunächst aber gezielt verzichtet. Nachdem so der Aufwand bei Neuerstellung eines Modells abgeschätzt und auch typische „Einsteiger-Fallstricke“ erkannt werden konnten, ist das fertig erstellte Modelica-Modell in Hinblick auf Verwendung von vorliegenden Standard-Elementen modifiziert worden. Zum Einsatz kamen dabei die Modelica-Standard Bibliothek sowie das ThermoFluid-Paket. Dieses Vorgehen entspricht zwar ausdrücklich nicht den einschlägigen Empfehlungen, von vornherein möglichst viele vorhandene Elemente zu nutzen, es bietet jedoch bei mehr Arbeitsaufwand die beste Möglichkeit, alle interessierenden Aspekte zu beleuchten.

3 Grundlagen der objektorientierten Modellbildung mit Modelica

Zentrale Elemente der Modelica-Modelle sind die Schnittstellen (connector). Dort werden sämtliche Größen festgelegt, die Systemgrenzen überschreiten sollen. Dabei wird im Gegensatz zur signalorientierten Modellbildung zwischen zwei Signaltypen unterschieden: Potentialgrößen sind diejenigen Größen, die an Verknüpfungspunkten immer gleich sind, typischerweise physikalische Zustände (Tabelle 1, links), Flussgrößen lassen sich durch eine „Nullsumme“ als Erhaltungssatz beschreiben (Tabelle 1, rechts). Diese unterschiedliche Beschreibungen sind fest in Modelica vorgegeben. Indem eine Variable als Instanz eines bestimmten connectors definiert wird, werden die entsprechenden Gleichungen automatisch ohne weiteren Benutzereingriff eingebunden.

Potential-Variablen:		Fluss-Variablen:	
<i>alle identisch:</i>	$var_1 = \dots = var_n$	<i>Erhaltungssatz als „Nullsumme“:</i>	$\sum var_i = 0$
Druck p		Massenstrom MF	(Massenbilanz)
Drehzahl n		Moment M	(Drallsatz)
Weg x		Kraft F	(Satz von Newton)
El. Spannung U		Elektrischer Strom I	(Kirchhoff'sche Knotenregel)

Tabelle 1: Beispiele für Potential- und Fluss-Variablen

Nach Definition aller Schnittstellen werden im equation-Abschnitt des Modells alle

vorhandene Größen mathematisch verknüpft. Dabei spielt es keine Rolle, wie vorliegende Gleichungen „sortiert“ werden, der Benutzer kann sämtliche Ausdrücke in beliebiger - auch impliziter - Form vorgeben.

4 Aufbau von Teilmodellen am Beispiel des AGR-Kühlers des Dieselmotors

Erstellte Teilmodelle lassen sich durch das Verknüpfen der Schnittstellen beider Elemente mittels eines Befehls oder gra`schen Verbindung zu einem übergeordneten Element verbinden. Dies kann sowohl direkt auf Code-Ebene (Modelica) als auch in der gra`schen Ober`ache geschehen. Beide Varianten sind einander in Abbildung 1 gegenübergestellt. Durch die Wiederverwendbarkeit einmal erstellter connectoren und Teilmodelle wird so der Modellaufbau deutlich vereinfacht.

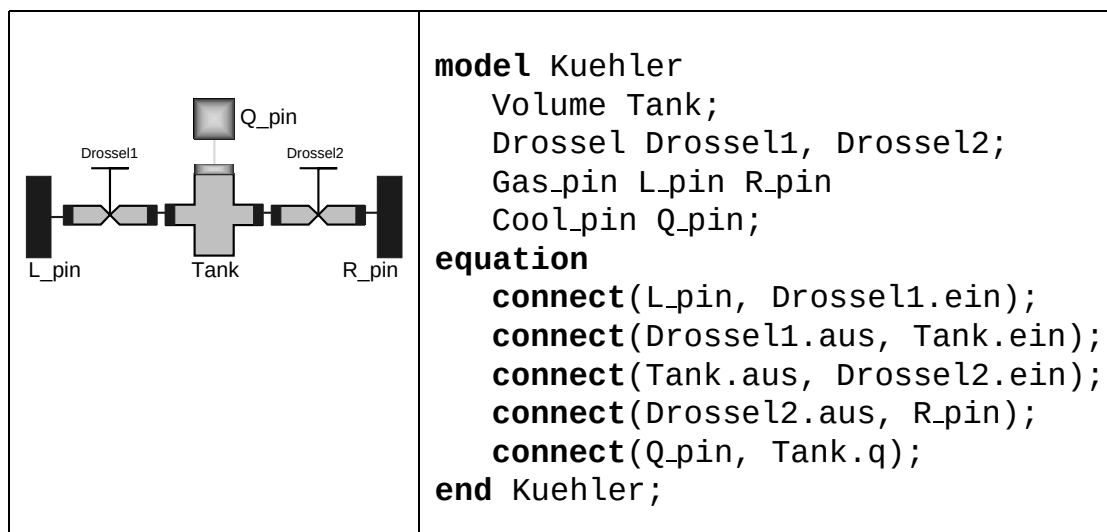


Abbildung 1: Zusammengesetzter Kühler - links Dymola-Ober`ache, rechts Modelica-Code

Dieses Modell kann ebenfalls eigene Schnittstellen aufweisen, um so wiederum in weitere Hierarchien eingebunden zu werden (*L_pin* und *R_pin*). Modelle aus vollständig funktionierenden Teilmodellen aufzubauen wird somit, dem Anspruch folgend, deutlich vereinfacht. Der hier gezeigte Ladeluftkühler besitzt neben der Möglichkeit zur Wärmeabfuhr (*Q_pin*) zwei identische Drosseln und ein Speichervolumen. Für die Drosseln liegt es daher nahe, zwei Instanzen derselben Bauteile (Objekte) mehrfach zu verwenden.

Betrachtet man nun die Umsetzung desselben Modells in Simulink (Bild 2), so stellt man fest, dass neben den auffälligen Rückführungen auch zwei verschiedene Umsetzungen der Drossel notwendig sind. Die Struktur des Modells fordert für die linke Drossel den Eingangsdruck als Ausgangsgröße, für die rechte ist der Massenstrom der Ausgang. Jede signalorientierte Darstellung erfordert daher eine eigene Umsetzung der Drosselgleichung für die unterschiedlichen Fälle.

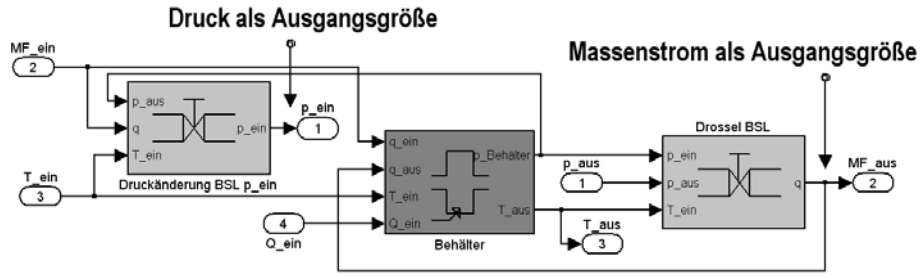


Abbildung 2: Kühler in Simulink

Existieren die einzelnen Elemente bereits, so ist die Modellierung in Dymola offensichtlich deutlich einfacher und durchgängiger als bei signal-orientierten Systemen. Wenn die zu Grunde liegenden Gleichungen komplexer werden, kann aber genau das Erstellen dieser Teilmodelle zu Schwierigkeiten führen. Neben vielen Komponenten, bei denen die übliche Vorgehensweise reibungslos funktioniert, zeigen sich gerade an der hier verwendeten Drosselgleichung für Modelica typische Probleme.

Gleichung 1 beschreibt den Zusammenhang zwischen Druckverhältnis Π und Massendurchfluss $\dot{m}$. Als weitere Konstanten sind zu sehen die Gaskonstante R und der Isentropenexponent κ , die beide nur vom durchströmenden Medium abhängen, der reduzierte Strömungsquerschnitt A_{red} sowie Druck und Temperatur des Fluids am Drosseleingang p_{ein} und T_{ein} . Sämtliche vom Druckverhältnis Π abhängigen Terme sind in dem Koeffizienten c_p zusammengefasst. Dessen Beschreibung ist nicht eindeutig umkehrbar, also nach Π auflösbar; für ein vorgegebenes c_p existieren eine Über- und eine Unterschalllösung für das Druckverhältnis. Es überrascht nicht, dass der Versuch, diese Gleichung „in beliebiger Form“ zu implementieren, auch mit geeigneter Initialisierung zunächst scheitert.

$$\dot{m} = \sqrt{\frac{2\kappa}{(\kappa - 1)R} \cdot A_{red}^2 \cdot \frac{p_{ein}^2}{T_{ein}} \cdot c_p}$$

$$\text{mit } c_p = \left(1 - \Pi^{\frac{\kappa-1}{\kappa}}\right) \Pi^{\frac{2}{\kappa}} \quad (1)$$

Die Umsetzung der einzelnen „gerichteten“ Drosseln in Simulink gestaltet sich hingegen relativ einfach. Wird der Massenstrom als Ausgang benötigt, kann Gleichung 1 ohne weitere Maßnahmen verwendet werden. Wird hingegen ein Druck als Ausgangsgröße gewünscht, bietet sich - zumindest für ein konstantes κ - die Offline Berechnung derselben Gleichung für den Unterschallbereich und Invertierung mittels einer fest abgelegten Tabelle an. Dieses Vorgehen ist prinzipiell auch in Dymola möglich, widerspricht jedoch dem grundlegenden Ansatz und ist nicht empfehlenswert.

Eine mögliche Lösung bietet eine geschlossene Umsetzung in der *ThermoFluid-Library* [9]. Wahlweise eine Verwendung eben dieser fertigen Bauteile aus der frei verfügbaren Bibliothek oder eine Anleihe bei zumindest einigen Teilen davon lassen so eine Lösung in der gewünschten Art zu.

Dies zeigt deutlich charakteristische Eigenschaften der Arbeit mit Modelica: Es gibt auch hier bestimmte Probleme, um deren mathematische Handhabung der Modellierer sich kümmern muss, wo also der Anspruch der Sprache, ihm alle notwendige Mathematik

abzunehmen, hinter der Wirklichkeit zurückbleibt. Dies gilt auch für einige sprachspezifische Besonderheiten, die berücksichtigt werden müssen, z.B. bei der Verarbeitung von Ereignissen, mit denen mögliche Unstetigkeiten behandelt werden. Sind diese Probleme aber einmal gelöst, existieren weitaus flexibler einsetzbare Modelle als dies bei Signalorientierung möglich ist. Hinzu kommt, dass solche Modelle innerhalb der Bibliotheken - zum großen Teil kostenlos - verfügbar sind und so im Idealfall einmal gelöste Probleme nicht erneut bearbeitet werden müssen.

Im Ergebnis führt dies alles dazu, dass mit einer geringen Einarbeitungszeit in Modelica/Dymola ein Modell des Motors aufgebaut werden konnte, das dem unter Matlab/Simulink ebenbürtig ist (Bild 3) - trotz des deutlich größeren Vorwissens der Autoren bezüglich Matlab. Für zukünftige Aufgaben bietet dieses Modell beste Voraussetzungen. Leichte Änderungen in der Motorkonfiguration, z.B. ein Tauschen von AGR-Kühler und AGR-Ventil („AGR-Ventil auf der heißen Seite“ bzw. „AGR-Ventil auf der kalten Seite“), können so deutlich einfacher vorgenommen werden, als dies unter Simulink der Fall ist.

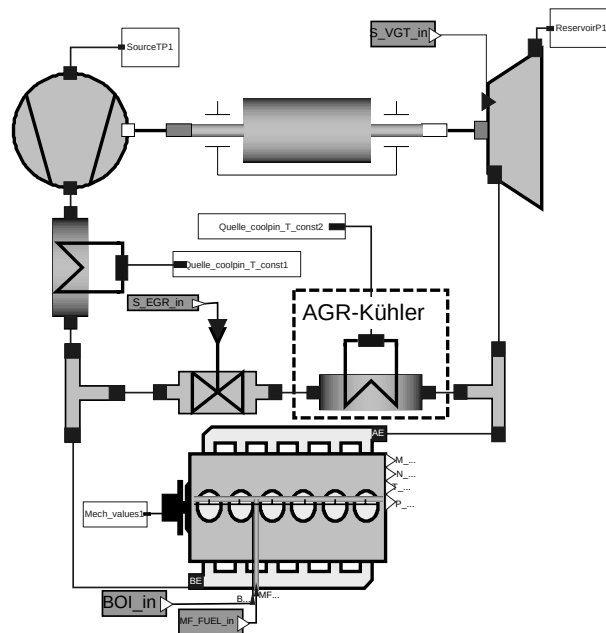


Abbildung 3: Vollständiger Motor in Dymola

5 Simulation und Ergebnisse

Die mit beiden Systemen erzeugten Simulationsergebnisse sind in der Abbildungsgenauigkeit vergleichbar. In Bild 4 ist ein simulierter Lastsprung am AGR-Kühler des Motors zu sehen. Temperatur- und Massenstromverlauf sind nahezu identisch, lediglich beim Druckverlauf sind Abweichungen deutlich sichtbar. Einerseits kann dies durch eine verwendete Näherung bei der Umsetzung der Drossel in der Thermo'uid-Bibliothek hervorgerufen sein (Abschnitt 4), andererseits werden aber auch unter Simulink häufig Hilfskonstrukte wie zusätzliche Memory-Blocks verwendet, um die Simulation zu stabilisieren. Auch

ein Vergleich mit dem mit deutlich höherem Detailgrad gerechneten Verlauf aus einem Prozessrechensystem lässt hier keine eindeutige Aussage zu.

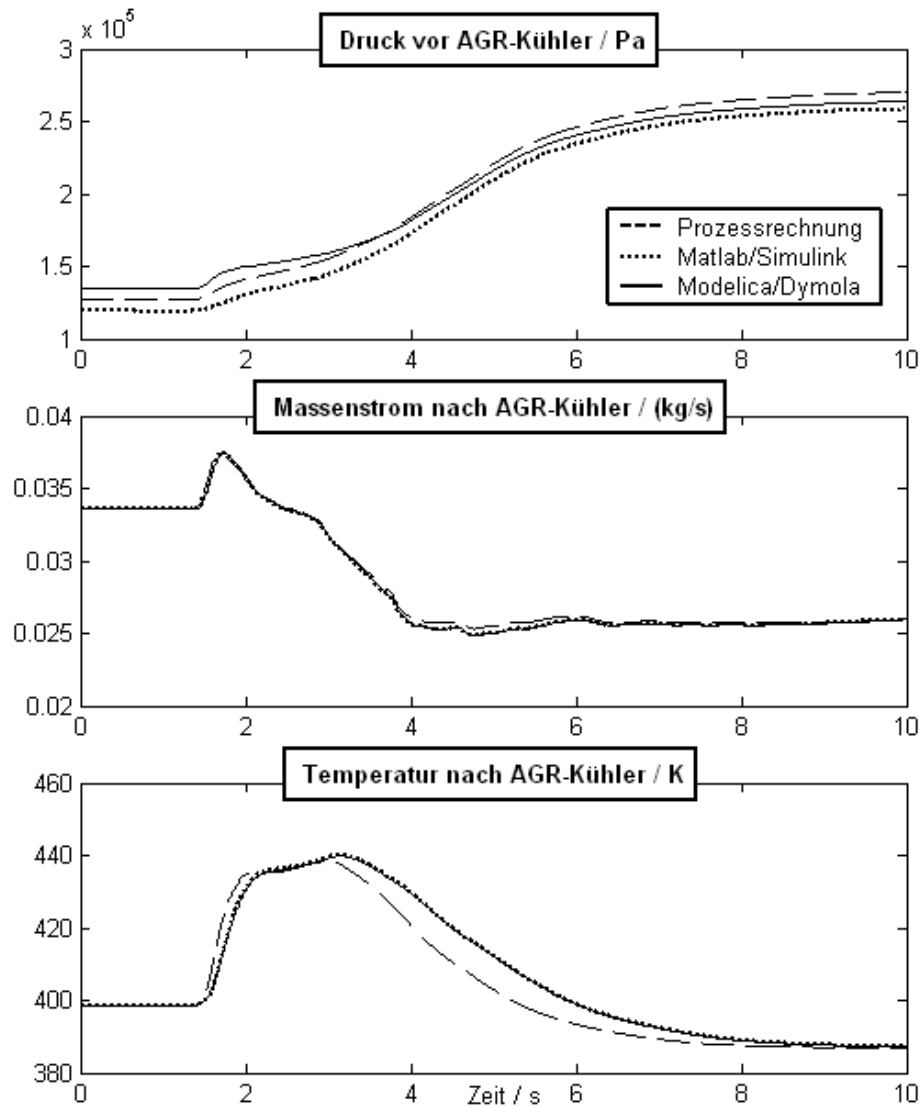


Abbildung 4: Simulationsergebnisse: Prozessrechnung, Simulink und Dymola

Deutlich größere Unterschiede zeigen sich bei Betrachtung der Rechenzeit und der Stabilität: Bei einem Runge-Kutta-Integrations-Algorithmus vierter Ordnung mit einer Schrittweite von einer Sekunde benötigt Simulink auf einem 1,4 GHz-System 63 Sekunden für eine Simulation von 10 Sekunden. Dymola benötigt für die gleiche Aufgabe lediglich 4,5 Sekunden. Dies verwundert jedoch nicht, wenn man bedenkt, dass Dymola vor der Simulation aus dem Modelica-Modell C-Code erzeugt und diesen kompiliert. Dieser Vorgang dauert beim vorliegenden Modell bis zu neun Sekunden und macht damit unter Umständen den größten Teil der Simulation aus. Auch mit Berücksichtigung dieser Zeit ist der Gesamt-Vorgang von Dymola jedoch schneller. Auch bei der Stabilität wirkt sich die vorherige Übersetzung positiv aus. Hier wird da eigentlich zu simulierende System neu erzeugt, wobei die Ordnung des Systems reduziert wird und algebraische

Schleifen analytisch aufgelöst werden. Deutlich tritt der Effekt dieses Vorgehens auf, wenn man das Dymola-Modell mit Hilfe des Dymola-Simulink-Interfaces in Simulink einbindet: Dymola bietet ein *Dymola-Simulink-Interface*, mit dessen Hilfe sich Dymola-Modelle einfach als s-function in Simulink einbinden lassen. Verglichen wurden hier also das ursprüngliche Simulink-Modell sowie das bereits compilierte und in Simulink eingebundene Dymola-Modell. Entgegen den Erwartungen zeigt die Dymola-Variante bei identischer fester Schrittweite keine Geschwindigkeitsvorteile, sondern läuft sogar etwa 20% langsamer. Allerdings benötigt die Simulink-Variante ein Schrittweite von weniger als 0,6 ms; darüber hinaus wird die Simulation instabil und die Ergebnisse werden stark verfälscht oder vollständig unbrauchbar. Die Dymola-Variante läuft hingegen bis zu einer Schrittweite von 10 ms stabil, so dass damit im Endeffekt ein Geschwindigkeitsvorteil von einer Zehnerpotenz erzielt wird! Auch eine mittels des Realtime-Workshops aus dem Simulink-Modell erstellte s-function lässt sich nicht mit höherer Schrittweite als das ursprüngliche Simulink-Modell betreiben und zeigte keine nennenswerten Geschwindigkeitsverbesserungen. Wie schon der Einsatz der Dymola-Variante vermuten ließ, scheint auch hier der Vorteil der schnelleren Ausführbarkeit des compilierten C-Codes durch den für s-functions notwendigen Code-Overhead ausgeglichen zu werden.

Durch die gute Handhabbarkeit und die höhere Stabilität der Dymola-basierten Modelle ist damit auch gerade der kombinierte Einsatz von Simulink- und Dymola-Modellen unter Matlab eine interessante Alternative. Dies wird besonders unterstützt durch die bequeme Handhabung des *Dymola-Simulink-Interface*, das sich nahtlos als neuer Block in die Simulink-Umgebung einfügt. Ist solch ein Block eingefügt und mit einem Dymola-Modell verknüpft, lässt sich aus Simulink heraus das Modell in Dymola öffnen, bearbeiten und neu übersetzen. So bleibt für den Anwender die Funktionalität beider Programme vollständig erhalten.

6 Handhabung und Auswertung

Neben der eigentlichen Modellbildung gibt es zusätzliche Anforderungen an die Handhabung des Modells. Dies betrifft vor allem die Visualisierung der Ergebnisse und Adaption der Modelle an verschiedene Szenarien, Arbeitspunkte etc. In Matlab müssen alle darzustellenden Signale explizit vor dem Beginn der Simulation festgelegt werden. Die Ausgabe erfolgt über separate Blöcke, z.B. *Scopes*. In Dymola hingegen stehen nach der Simulation automatisch alle Signale zur Verfügung und können frei zur Darstellung ausgewählt werden. Dies stellt zwar deutlich höhere Speicher-Anforderungen, ist jedoch sehr komfortabel. Nachteilig wirkt sich dabei jedoch aus, dass einmal festgelegte Plotkonfigurationen sich nicht speichern lassen. Auch die Möglichkeiten, diese Visualisierung skriptgesteuert auszuführen, bleibt weit hinter den Möglichkeiten von Matlab zurück.

Insgesamt ist die Bediener-Schnittstelle und die Anpassungsfähigkeit derselben sicherlich der Bereich, in dem weitere Verbesserungen in Dymola wünschenswert sind. Die Modelle selbst sind voll parametrierbar und dynamisch mit Eingangsdaten zu versehen, jedoch fehlt vollständig die Möglichkeit, wie in Matlab (z.B. mit *GUIDE*) eigene Bedienoberflächen zu erstellen. Auch bezüglich Auswertung, Analyse, Reglerentwicklung und zahlreichen weiteren in Matlab zur Verfügung stehenden Werkzeugen und

Toolboxen für die verschiedensten Bereiche hat Dymola nichts entgegenzusetzen. Diese Ergänzungen sind zwar nicht ursächlich für den Modellaufbau und die Simulation notwendig, machen aber gerade die Einbindung in andere Arbeitsschritte deutlich einfacher oder überhaupt erst möglich.

Dieser Problematik ist man sich aber bei Dymola durchaus bewusst. Dymola tritt jedoch nicht in Konkurrenz zu Matlab, sondern bietet sinnvolle Erweiterungen zur Kombination der Programme. So speichert Dymola die Simulationsergebnisse im von Matlab lesbaren *mat*-Format und wie in Abschnitt 4 erläutert wurde, lassen sich mit Hilfe des *Dymola-Simulink-Interfaces* ganze Modelle in Simulink einbinden. Bei komplexen Modellen kann es auch hier zu notwendigen Feinarbeiten kommen, insgesamt bietet dies jedoch den interessanten Ansatz, die Fähigkeiten von Modelica/Dymola bei der Modellbildung mit der Mächtigkeit von Matlab nicht nur für der Bedienung und Auswertung zu kombinieren, sondern auch die umfangreichen Möglichkeiten Matlabs zur Analyse und Synthese für die Reglerentwicklung zu nutzen.

7 Fazit

Nach den bisherigen Erfahrungen kann es nach Ansicht der Autoren nicht um den Alleinherrschaftsanspruch eines Werkzeuges gehen. Vielmehr zeigt sich, dass - wie so oft - eine sinnvolle Kombination der Stärken unterschiedlicher Ansätze sinnvoll ist. Modelica hat nicht ohne Grund innerhalb der letzten Jahre einen hohen Bekanntheitsgrad erlangt und bietet eine gute Möglichkeit, Modellbildung zu vereinheitlichen und zu standardisieren. Der Simulator Dymola kann seine Stärken dort voll ausspielen, wofür er entwickelt wurde: Beim grafischen Erstellen der Modelle und beim schnellen und numerisch stabilen Simulieren derselben. Bei den Zusatzwerkzeugen hat Matlab einen über viele Jahre gewachsenen Vorsprung, den zu verringern nach derzeitigem Stand nicht Hauptziel der Entwickler von Dymola zu sein scheint. Vielmehr profitiert der Anwender von einer guten Zusammenarbeit beider Programme und erhält nebenbei gerade durch den ständigen Wechsel zwischen den Herangehensweisen einen weiteren Blickwinkel und größeres Systemverständnis.

Literatur

- [1] *JaLa* am IRT der RWTH Aachen: <http://www.irt.rwth-aachen.de>
- [2] H. Elmqvist, S. Mattson, "An Introduction to the Physical Language Modelica". *Proceedings of the 9th European Simulation Symposium (ESS 97)*, 1997.
- [3] M. Otter, "Objektorientierte Modellierung Physikalischer Systeme Teil 1-5". *Automatisierungstechnik (at)*, 47. Jahrgang, Heft 1-4, Oldenbourg Verlag, München (1999).
- [4] J. Rückert, F. Richert, A. Schloßer, D. Abel, "Konzepte zur Regelung von Ladedruck und AGR-Rate beim Nutzfahrzeug-Dieselmotor". 6. GMA-Kongress, 03-04.06. 2003, Baden-Baden.

- [5] A. Pfeifer, M. Smeets, H.-O. Herrmann, D. Tomazic, F. Richert, A. Schloßer, "A New Approach to Boost Pressure and EGR Rate Control Development for HD Truck Engines with VGT". *SAE World Congress 04.-07.03.2002, Detroit, SAE 2002-01-0964*.
- [6] F. Richert, A. Schloßer, "Modellbildung eines Schwerlastdieselmotors mit Abgasrückführung und Turbolader mit variabler Turbinengeometrie". *Workshop ASIM-Fachgruppe 4.5.1, 04./05.03. 2001, Bielefeld*.
- [7] A. Schloßer, "Modellbildung und Simulation zur Ladedruck- und Abgasrückführregelung an einem Dieselmotor". *Fortschritts-Berichte VDI, Reihe 8, Nr. 860, VDI-Verlag, Düsseldorf, 2000*.
- [8] M. Tiller, "Introduction to Physical Modeling with Modelica". *The Kluwer international series in engineering and computer science, SECS 615, Dordrecht, 2001*.
- [9] Verfügbare Bibliotheken für Modelica, <http://www.modelica.org/libraries.shtml>

Entwicklung eines steuergerätefähigen Modells für einen Drei-Wege-Katalysator

P. Gieseemann, IAV GmbH

Zusammenfassung

Vorgestellt wird ein physikalisches Drei-Wege-Katalysatormodell zur Modellierung des Luft-Kraftstoff-Verhältnisses Lambda hinter Katalysator, welches unter der strengen Anforderung der Steuergerätefähigkeit entwickelt wurde. Als Haupteinflussgrößen des Modells werden zunächst Lambda vor Katalysator und der Luftmassenstrom betrachtet. Grundlage für das Modell war ein aus der Literatur bekannter Modellansatz, welcher sukzessiv verändert und erweitert wurde.

Der Ansatz basiert auf einem dynamischen Sauerstoffspeichermodell, da das Katalysatorverhalten im direkten Zusammenhang mit der Fähigkeit steht, Sauerstoff einzuspeichern. Bei dem Modell wird zwischen Sauerstoffaufbau- und -abbaureaktionen unterschieden. Der ursprünglich lineare Ansatz des Sauerstoffspeichermodells bezüglich Lambda vor Katalysator wurde im Zuge der Untersuchungen durch ein nichtlineares Modell ersetzt. Des Weiteren wurde ein Teilmodell zur optimalen Bestimmung des Umschaltzeitpunktes zwischen Sauerstoffaufbau und -abbau eingefügt. Die zuvor unplausiblen Modelleigenschaften beim Wechsel von Sauerstoffaufbau zu -abbau und umgekehrt, die aus der bisherigen Ermittlung des Umschaltzeitpunktes resultierten, konnten somit korrigiert werden. Ebenso wurde der Eingangsraum durch die zeitliche Änderung von Lambda vor Katalysator erweitert, da dieser vor allem den Umschaltzeitpunkt zwischen Sauerstoffaufbau und -abbau beeinflusst und damit das Modellverhalten wesentlich verbessert. Die Berechnung der maximalen Sauerstoffspeicherfähigkeit des Katalysators wurde weitestgehend übernommen. Sie beruht auf dem Ausmessen der Flächen zwischen den Verläufen von Lambda vor und hinter Katalysator, die durch die Anregung von Lambdasprüngen zu beobachten sind. Die maximale Sauerstoffspeicherfähigkeit ist arbeitspunktabhängig und wird somit für verschiedene Luftmassenströme identifiziert.

Das resultierende Katalysatormodell zeigte in den ersten Anwendungen, bei dem ein Front-Katalysator zugrundegelegt wurde, eine hohe Approximationsgüte des Lambdas hinter Katalysator. Aufgrund der geringen Modellkomplexität und der hohen Genauigkeit liefert das Modell somit die Basis einer Echtzeit-Vorhersage für das Katalysatorverhalten. In einem nächsten Entwicklungsschritt ist geplant, das Modell um weitere Einflussgrößen wie Temperaturen und Alterung des Katalysators zu erweitern. Insgesamt bietet eine Vorhersage des Katalysatorverhaltens im laufenden Fahrbetrieb folgende Vorteile: Eine verlässliche Vorhersage des Lambdas hinter Katalysator würde die Einsparung der Lambda-Sonde hinter Katalysator ermöglichen und demzufolge eine Kostenreduzierung bewirken. Ein weiterer Vorteil ergibt sich in Bezug auf die Schadstoffminimierung, da mithilfe des Katalysatormodells frühzeitig in die Lambdae Regelung eingegriffen werden könnte. Schließlich sind Einsatzmöglichkeiten im Bereich der On-Board-Diagnose oder im Zusammenhang mit einem NO_x -Speicherkatalysator denkbar.

Modellbildungsstrategien und Berechnungsmethodik in der Abgasanlagenentwicklung

S. Streck¹; J. Smirek²; V. Fäßler³

Kurzfassung:

Kurze Entwicklungszeiten, Variantenvielfalt und hohe Anforderungen an Qualität und Lebensdauer machen den Einsatz moderner Simulationstools in der Abgasanlagenentwicklung unverzichtbar. Am praktischen Beispiel einer typischen PKW-Abgasanlage wird die Vorgehensweise der Simulation mittels FE-Berechnung dargestellt. Die Modellierung der Abgasanlage ist wesentlicher Bestandteil der Simulation und entscheidet über die Aussagekraft der erzielten Ergebnisse. Die statische Berechnung der Abgasanlage unter Eigengewicht gibt Aufschlüsse über das Lagerungskonzept. Anhand des berechneten Schwingungsverhaltens erfolgt die Verifikation des Modells mit der Betriebsschwingungsanalyse am Prüfstand. Die im Anschluss folgende Spannungsanalyse in den Resonanzfrequenzen zeigt Optimierungspotentiale der Anlage hinsichtlich ihrer Festigkeit auf. Die Erkenntnisse der Berechnungen fließen unterstützend in den Konstruktionsprozess ein.

Abstract:

Short development times, variants diversity und high demands on quality and product life span make the use of modern simulation tools indispensable. The procedure of simulation using FE calculation is exemplified through a modern exhaust system. The FE - modeling of the exhaust system is an integral part of the simulation and decides on the significance of the obtained results. The static calculation of the exhaust system under deadweight gives information about the suspension concept. On the basis of the calculated vibration behavior the model is

¹ TWT GmbH NL Friedrichshafen. FB Engineering, Team Structure Analysis 1.
stephan.streck@tw-t-gmbh.de

² TWT GmbH Neuhausen. FB Engineering. josef.smirek@tw-t-gmbh.de

³ TWT GmbH NL Friedrichshafen. Technology & Research. victor.faessler@tw-t-gmbh.de

verified with the vibration analysis at the test bench. The following stress analysis for the resonant frequencies reveals potentials for optimization of the system with respect to its strength. The calculated findings exert supporting influence on the design process.

1 Einführung

Die Entwicklung neuer Produkte im Automobilbau hat immer höheren Ansprüchen zu genügen. Die Anforderungen an die Bauteile werden komplexer und die Entwicklungszeiten sollen kürzer werden. Diese Entwicklung macht auch vor den Abgasanlagen nicht halt. Hinzu kommen hier die Variantenauslegungen für die zahlreichen Motor-Baureihen-Kombinationen und verschärfte Abgasnormen. Ohne den Einsatz von modernen Entwicklungsmethoden des CAE ist die Einhaltung von Entwicklungszeiten und Qualitätsanforderungen nicht zu realisieren.

Mit Hilfe von FE-Berechnungen können bereits Aussagen über thermodynamische, statische und dynamische Belastungen gemacht werden, bevor ein Prototyp vorliegt. Aufwendige Prüfstandsversuche und Dauerlaufprüfungen können so auf ein Minimum reduziert werden.

Am praktischen Beispiel einer zweiflutigen Abgasanlage mit einem V8-Motor werden die wesentlichen Schritte der Simulation dargestellt. Auf der Grundlage von CAD-Daten erfolgt die Modellerstellung. Nach Aufbringen der Randbedingungen wird das Modell verifiziert. Die Erkenntnisse aus den Ergebnissen der statischen Berechnungen, der Schwingungsanalyse sowie der Spannungsanalyse fließen in den Konstruktionsprozess ein und werden vom Konstrukteur zur Optimierung genutzt. Im Anschluss an die Simulationsaufgabe wird die optimierte Anlage im Dauerlauf überprüft. In Abbildung 1 sind die Phasen der Simulation dargestellt.

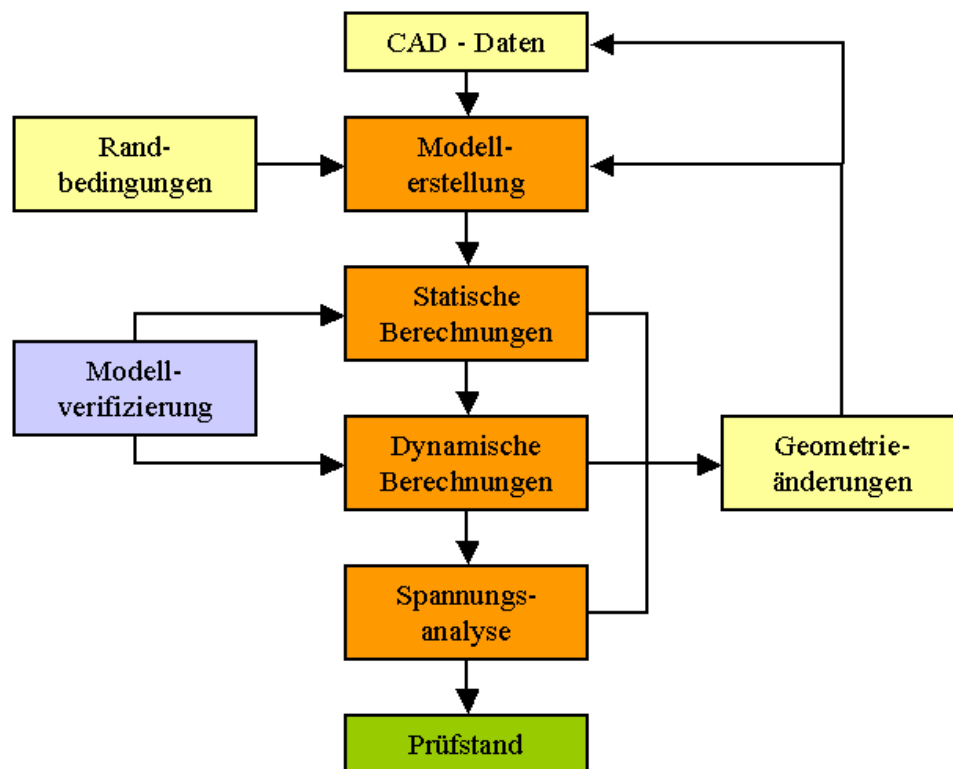


Abbildung 1 - Ablauf der Simulation

2 Modellbildung

Grundlage der Berechnung und Simulation ist ein FE-Modell mit physikalischen Eigenschaften der realen Abgasanlage. Die Modellbildung erfolgt auf der Basis von 3D-CAD-Daten mit Hilfe eines speziellen Preprozessors.

Hybride FE-Modelle sind Voraussetzung für eine zufriedenstellende Abbildung der spezifischen Eigenschaften der Abgasanlage. Schalenelemente repräsentieren alle Blech- bzw. Rohrteile, und mit Volumenelementen werden Flanschverbindungen, Katalysatorblöcke und Schweißnähte abgebildet. Eigenschaften von Leitungen oder Gummielementen werden als Massepunkte oder Federelemente dargestellt. Der

Motor-Getriebeverbund wird als Starrkörpermodell abgebildet, in dem die Massen, Trägheits- und Deviationsmomente berücksichtigt werden.

Ein wesentliches Kriterium bei der Modellbildung ist die Wahl des Detaillierungsgrades. Eine niedrige Detaillierung kann zu ungenauen Ergebnissen führen, eine hohe Detaillierung führt zu einem unnötig hohen Modellierungs- und Rechenaufwand.

Wurden in den vergangenen Jahren noch viele Einzelheiten vernachlässigt oder vereinfacht, werden heute fast alle Details der Abgasanlage bis hin zu den karosserieeitigen Haltern modelliert.

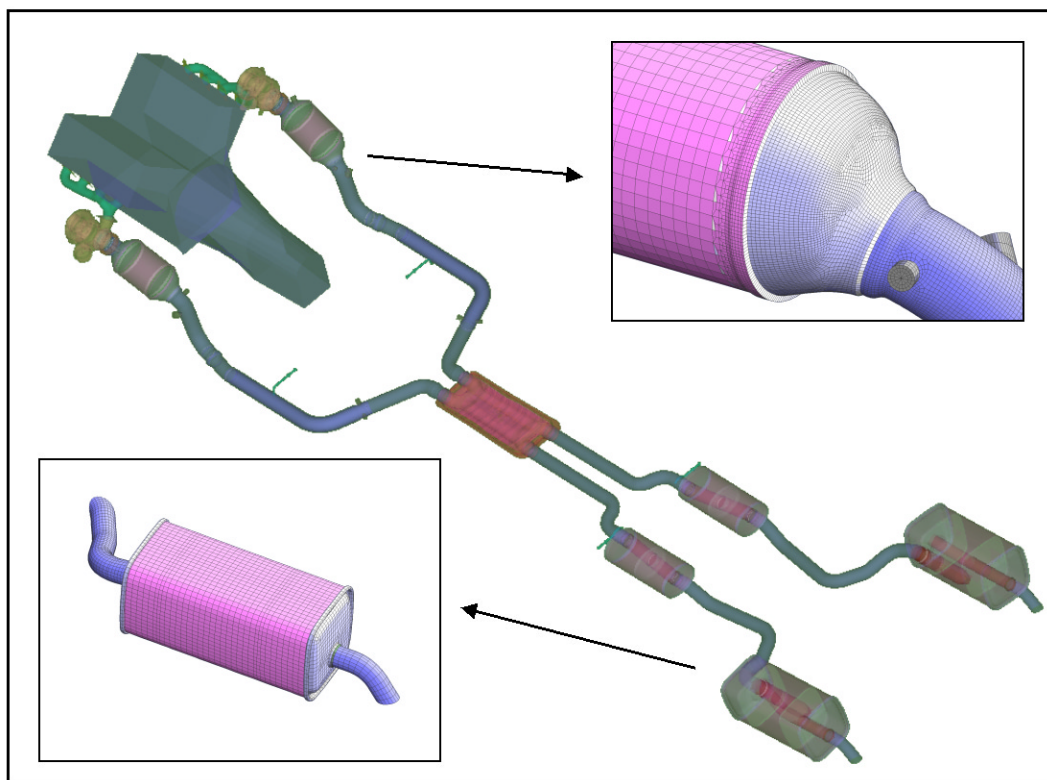


Abbildung 2 - Modell der Abgasanlage

Das FE-Modell ist modular organisiert und vereinfacht die Variantenuntersuchungen bei geänderten Halterpositionen, Wandstärken oder Rohrverläufen. Eine

Anpassung der Abgasanlage an andere Motoren ist unter Verwendung neuer Krümmermodelle in kurzer Zeit möglich.

Die Modellierung von Bauteilverbindungen, etwa Verschweißungen oder Verschraubungen erfordert spezielle Kenntnisse, die durch Abgleich von Simulation und Experiment gewonnen wurden. Werden die Verbindungen bezüglich ihrer Steifigkeit zu hart oder zu weich abgebildet führt dies zu Simulationsergebnissen, die möglicherweise numerisch korrekt sind, jedoch das Verhalten der simulierten Abgasanlage nicht richtig beschreiben.

Nach der geometrischen Modellierung werden den Elementen ihre spezifischen Eigenschaften wie Blechdicken, Querschnittswerte von Stabelementen und Federkennwerte zugewiesen. Zudem werden die Materialeigenschaften eingetragen.

Für die Berechnungen der natürlichen Schwingungsfrequenzen und –formen ist ein relativ grobes Modell ausreichend, für die Auflösung der Spannungen benötigt man allerdings ein feines Netz, gerade in Bereichen, in denen hohe Spannungsgradienten zu erwarten sind. Diese Bereiche sind vor allem Querschnittssprünge bei Katalysatorein- und -ausgängen, sowie an Stellen von Krafteinleitungen oder an Aufhängungen. Trotz dieser unterschiedlichen Anforderungen an das Modell wird lediglich ein Modell für die gesamten Berechnungen erstellt. Die Erstellung eines Grobmodells und die spätere Verfeinerung würde mehr Zeit in Anspruch nehmen als die Erstellung eines Modells, das allen Simulationen genügt.

In den letzten Jahren hat aufgrund eines steigenden Detaillierungsgrades die Anzahl der Knoten und Elemente des Gesamtmodells kontinuierlich zugenommen. Je nach Anlagentyp ergeben sich derzeit Modellgrößen von 150.000 - 300.000 Elementen.

3 Randbedingungen

Als Randbedingungen werden Lagerungen und Gravitation, sowie Anregung durch den Motor oder Fahrbahneinflüsse berücksichtigt. Die Motoranregung setzt sich aus pulsierenden Gaskräften und freien Massenkräften und –momenten zusammen.

Die Temperaturabhängigkeit der verwendeten Materialien wird in den statischen und dynamischen Lastfällen berücksichtigt. Dazu wird durch eine vorausgehende thermische Analyse die Temperaturverteilung an der Abgasanlage ermittelt.

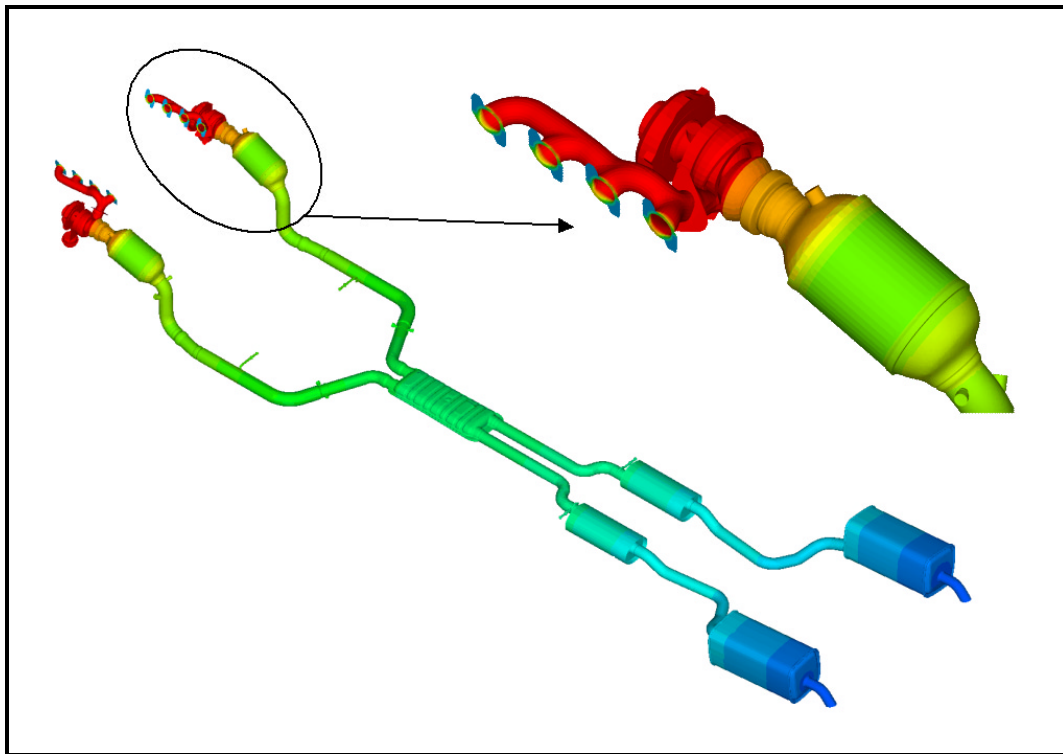


Abbildung 3 - Temperaturverteilung bei Volllast

4 Verhalten unter Eigengewicht

Bei der Berechnung der Absenkung und Durchbiegung der Abgasanlage unter Eigengewicht kann das Aufhängungskonzept überprüft werden. Zudem werden die Kräfte an den Halterungen ermittelt. Ungünstige Halterpositionen und eine daraus resultierende ungleichmäßige Lastverteilung können festgestellt und korrigiert werden.

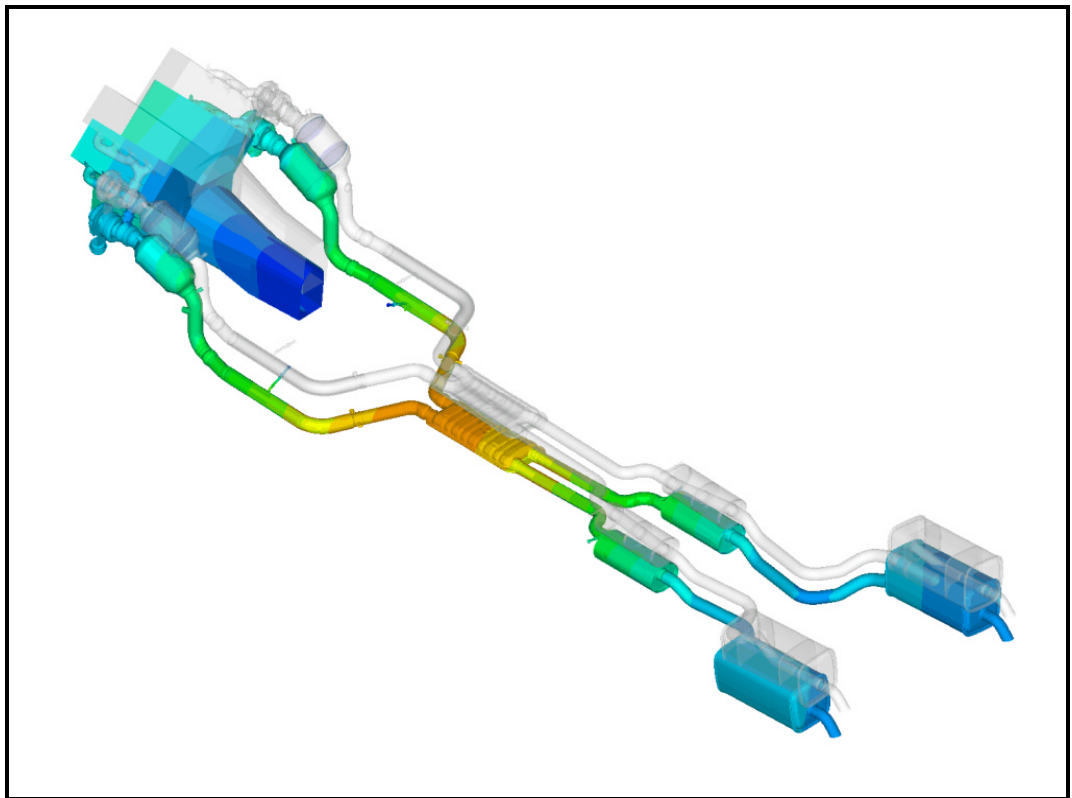


Abbildung 4 - Verschiebungen durch Eigengewicht

Die berechnete Absenkung der Anlage lässt erste Aussagen über die Eignung der verwendeten Federn oder Gummielemente zu. Anhand dieser Ergebnisse kann das Modell zudem mit gemessenen Absenkungsdaten verifiziert werden, sofern ein Prototyp der Abgasanlage vorliegt.

Die Durchbiegung liefert weitere Anhaltspunkte für die Überprüfung des Lagerkonzeptes.

5 Simulation des Schwingungsverhaltens

Die Berechnung des Schwingungsverhaltens wird in zwei Schritten durchgeführt. Zunächst werden mit Hilfe der Modalanalyse die natürlichen Schwingungsfrequenzen und -formen der Abgasanlage berechnet.

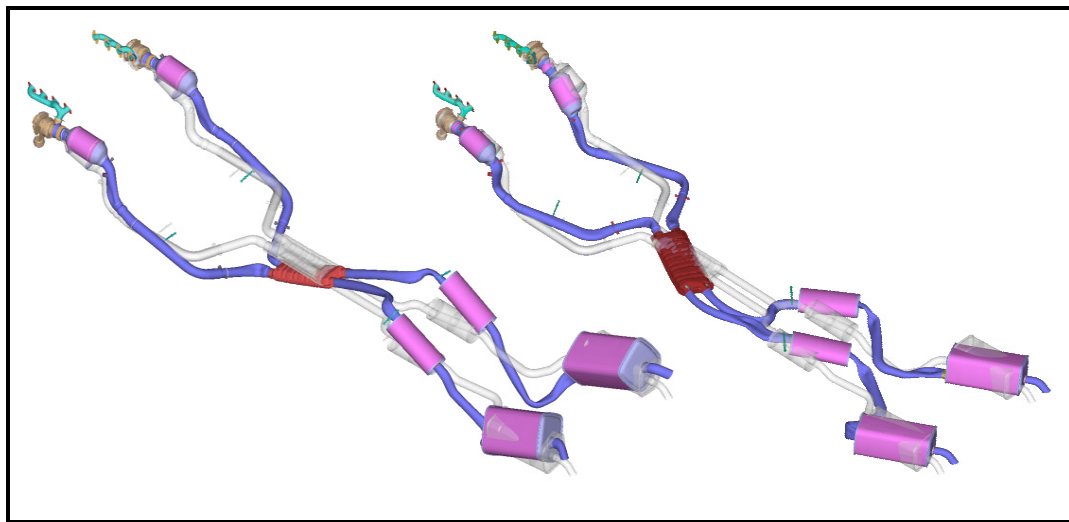


Abbildung 5 - Schwingformen

Auf der Basis der ermittelten Moden erfolgt die Berechnung des realen Schwingungsverhaltens im Nulllast- bzw. Volllastbetrieb infolge der äußeren Anregung.

Im vorliegenden Beispiel wird die Belastung durch einen V8-Motor simuliert, dessen größte Anregung von der 4. Motorordnung verursacht wird. Diese Anregung wird als konstantes Moment über die Drehzahl des Volllastbereiches auf das Motormodell aufgebracht.

Abbildung 6 zeigt an ausgewählten Positionen der Abgasanlage die Wegamplituden als Funktion über die Frequenz aufgetragen. Die Spitzen der Amplituden zeigen, in welchem Maße die Resonanzen im Frequenzbereich angeregt werden. Für ausgewählte Resonanzfrequenzen wird die Spannungsanalyse durchgeführt, da hier die höchsten Spannungen zu erwarten sind.

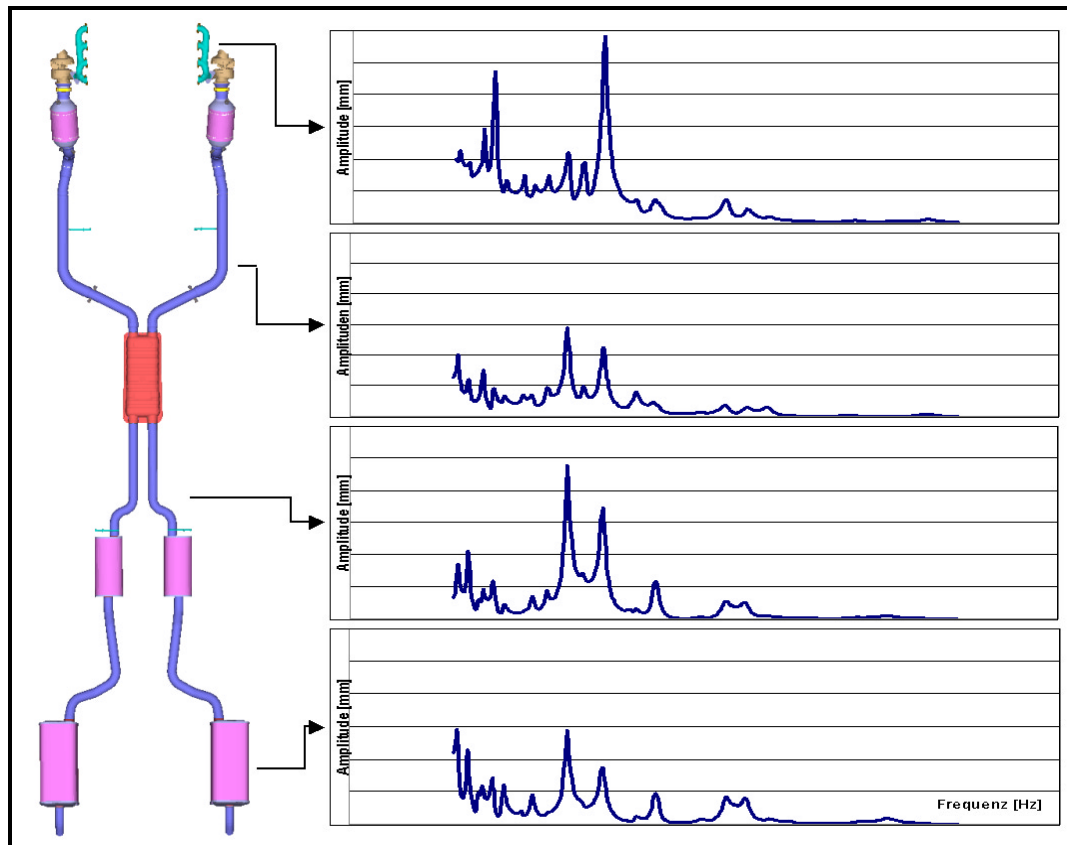


Abbildung 6 - Absolutverschiebungen über Frequenzen an ausgewählten Positionen

Mit Hilfe der Ergebnisse einer experimentellen Betriebsschwingungsanalyse kann das FE-Modell verifiziert und abgeglichen werden, um eine höhere Genauigkeit der in der Spannungsanalyse ermittelten Absolutwerte zu erzielen. Wenn zu diesem Zeitpunkt der Simulation kein Prototyp vorhanden ist, wird die Modellgültigkeit mittels Analogiebetrachtungen zu ähnlichen, bereits verifizierten Abgasanlagen überprüft, was eine weitreichende Erfahrung des Projektengineurs voraussetzt.

6 Spannungsanalyse

Die Spannungsberechnung erfolgt mit der Frequency Response Analyse bei den ermittelten Resonanzfrequenzen. Als Anregung dient in diesem Falle der Momentenverlauf der 4. Motorordnung.

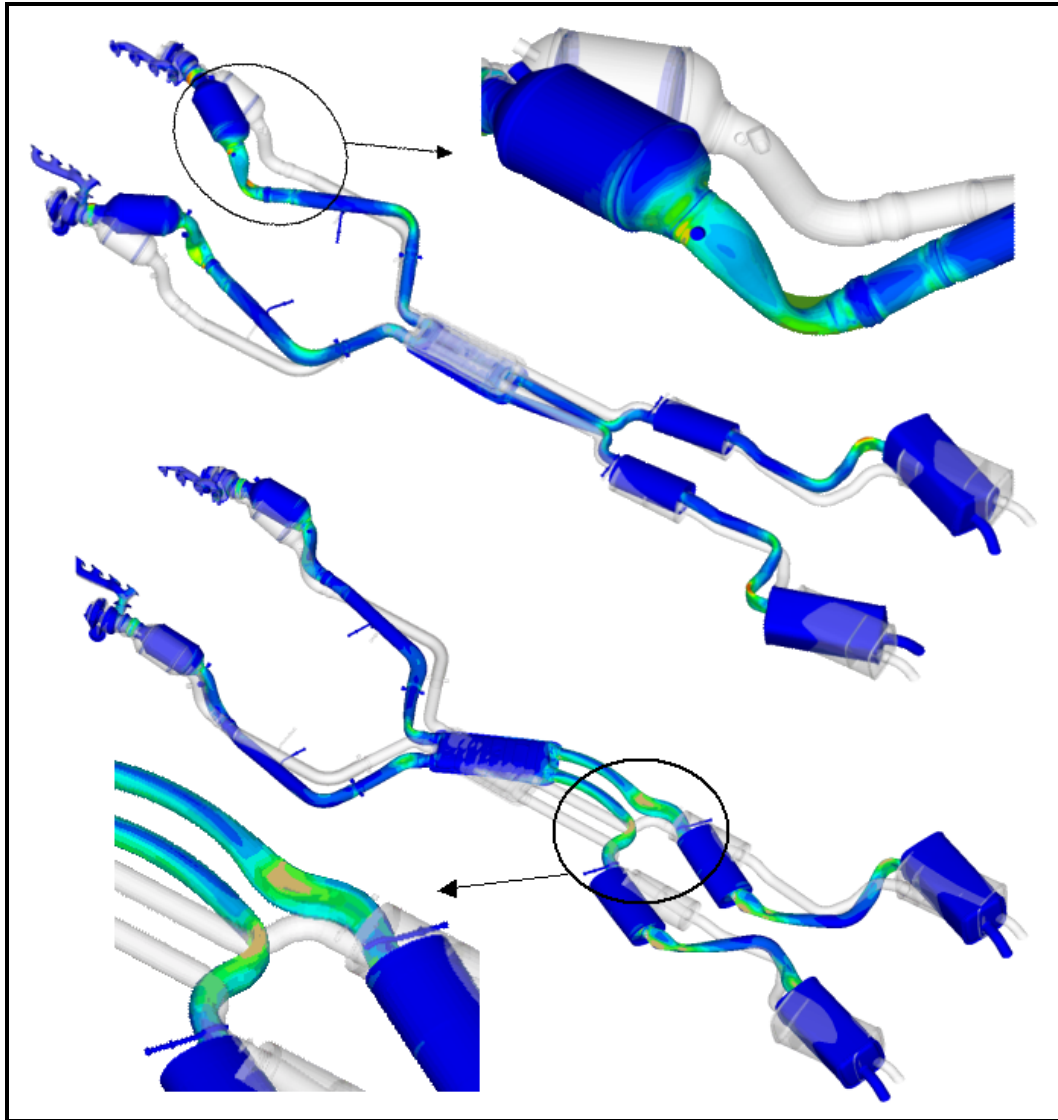


Abbildung 7 – Vergleichsspannungen bei Resonanzfrequenzen

Hohe Spannungen treten vor allem in Bereichen von geometrischen Kerben und an Übergängen von Rohren in Katalysatoren und Schalldämpfern auf. Konstruktive

Änderungen wie das Anbringen von Streben können hier eine Spannungssenkung herbeiführen.

Durch den Einsatz von Abkoppellementen oder Tilgern, deren Auslegung und Wirkung mit Hilfe der Simulation unterstützt und nachgewiesen wird, kann die Beanspruchung der Abgasanlage weiter gesenkt werden.

7 Zusammenfassung

Neben der Beanspruchung durch Temperaturbelastung sind Abgasanlagen auch dynamisch hoch belastete Systeme. Durch den Einsatz der FEM können Schwachstellen vor der Prototypenphase erkannt und Verbesserungspotentiale genutzt werden. Dadurch verringert sich die Entwicklungszeit für diese Systeme und Dauerlaufprüfungen können auf ein Minimum reduziert werden; im Idealfall wird nur eine optimierte Abgasanlage geprüft. Im Gegensatz zu Prüfstandsversuchen, bei denen Spannungen nur lokal gemessen werden können, zeigen Spannungsberechnungen durch FE-Simulation die Beanspruchung in allen Bereichen der Abgasanlage.

Eine sorgfältige Modellbildung nimmt den größten Zeitanteil der Simulationaufgabe in Anspruch und erfordert weitreichende Erfahrung. Ein FE-Modell, in welchem die wesentlichen physikalischen Eigenschaften berücksichtigt und realistische Randbedingungen angenommen wurden, ist die Grundlage dafür, genaue und realistische Ergebnisse des Schwingungsverhaltens zu erhalten. Modular organisierte Modelle erleichtern die Variantenberechnung.

Die Verifizierung des Modells erfolgt durch den Abgleich mit den Ergebnissen der experimentellen Betriebsschwingungsanalyse oder, falls kein Prototyp vorhanden ist, über Analogiebetrachtungen mit ähnlichen, bereits verifizierten Abgasanlagen und dient zur Erhöhung der absoluten Genauigkeit der Berechnungsergebnisse.

Literatur

- BOSSEL, H.: Modellbildung und Simulation, -Konzepte, Verfahren und Modelle zum Verhalten dynamischer Systeme-. Vieweg 1994
- BRASS, H.-H.; SEIFFERT, U.: Vieweg Handbuch Kraftfahrzeugtechnik. Vieweg 2001
- DRESIG, H.; HOLZWEIßIG, F.: Lehrbuch der Maschinendynamik. Springer 2003 i.D.
- HAGEDORN, P.: Über die Tilgung selbsterregter Schwingungen. ZAMP, Vol. 29, 1978, 815-821
- HEIL, B.; u.a.: Die Abgasanlage des Mercedes SL 500. MTZ 1/2002, Jg. 63, 6-14
- HÜSER, M. G.: Einsatz neuer Rechenverfahren in der Abgasanlagenentwicklung. Univ. Diss. TH Aachen 1999
- KATTGE, D.; SEFFLER, H.-W.: Die Bibliothek der Technik, Bd. 47, Abgasanlagen für Kraftfahrzeuge, Verlag Moderne Industrie 1990

Verwendung einer strömungsmechanischen Motorsimulation als 'virtueller Prüfstand' zur Untersuchung und Bewertung von Regelkonzepten am Beispiel eines PI-Gleichstellungsreglers für die Frischluftmassenströme in einem Biturbomotor

Dipl.-Ing. (FH) Micha Münzenmay, esip

Dipl.-Ing. (FH), MSc. Frank Hirsch, esip

Dr.-Ing. Hermann Kull, FHT-Esslingen

Einleitung

Im Zuge der Bemühungen zum Erreichen einer möglichst kurzen "time-to-market" spielen auch im Motorenbereich Simulationssysteme eine zunehmend bedeutendere Rolle. In diesem Zusammenhang ist es wünschenswert, sobald als möglich mit der Entwicklung von Regelungskonzepten und der Auslegung der Regelparameter beginnen zu können. Dazu wird ein Modell des Motors samt Luftsystem benötigt, um Simulationen durchführen zu können. Hiermit bietet sich auch die Möglichkeit problemlos Systemvariablen zu beobachten, deren Messung am realen Motor schwierig und dadurch teuer oder ungenau ist. Das größte Hindernis in dieser Entwicklung stellt sich in den Anforderungen an die Genauigkeit und Ablaufgeschwindigkeit der Motorsimulation. Modelle, die auf universellen Techniken zur Simulation technisch-physikalischer Prozesse beruhen, wie sie z.B. von Matlab/Simulink (Mathworks) zur Verfügung gestellt werden, sind mathematisch relativ kompakt und haben so den Vorteil sehr kurzer Ausführungszeiten. Ihr Nachteil liegt in der relativ aufwendigen Modellbildung und der oft nicht zufriedenstellenden Genauigkeit der Simulationsergebnisse. Die Präzision dieser Modelle kann zwar mit Hilfe von Systemidentifikations-Methoden z.T. erheblich verbessert werden, das ist aber erst möglich, sobald mindestens ein Exemplar des Motors gefertigt und entsprechend "vermessen" wurde. Spezialisierte eindimensionale finite Elemente Simulationen der Strömungsmechanik, wie in BOOST (AVL List GmbH) implementiert, ermöglichen eine relativ genaue Simulation eines Verbrennungsmotors inklusive Luftsystem, sobald alle relevanten Designparameter festgelegt sind. Der Nachteil dieser Art von Simulation ist der relativ hohe Rechenaufwand im Vergleich zur phänomenologischen Simulation. Aufgrund der stetig fortschreitenden Erhöhung der Rechengeschwindigkeit von Desktop-Rechnern verliert dieses Problem jedoch zunehmend an Relevanz.

BOOST wurde auch verwendet, um einen 6-Zylinder Biturbo-Dieselmotor mit einer neuartigen Luftsystemarchitektur zu modellieren. Die Besonderheit im Luftsystem dieses Motors besteht in der Zusammenführung der Frischluftmassenströme beider Turbolader, um einen identischen Ladedruck auf beiden Zylinderbänken zu garantieren. Dies führt jedoch zu dem Problem, daß die Massenströme der Turbolader gleichgeregelt werden müssen. Das ist nötig, um zu vermeiden, daß ein Turbolader vom anderen über die Pumpgrenze "gedrückt" wird. Mit Hilfe dieses Modells wurde untersucht, ob ein zu

diesem Zweck eingesetzter PI-Regler durch das Hinzufügen eines neuronalen Prädiktors verbessert werden kann.

Das Motormodell und BOOST

Das Modell, das zum Einsatz kam, wurde für Untersuchungen zur Eignung von eindimensionalen strömungsmechanischen finite Elemente Simulationen als "virtueller Prüfstand" für den frühen Test von Motorregelungen entworfen [Münzenmay02]. Es beschreibt einen 6-Zylinder Dieselmotor, der über zwei Turbolader mit variabler Turbinengeometrie (VTG) verfügt.

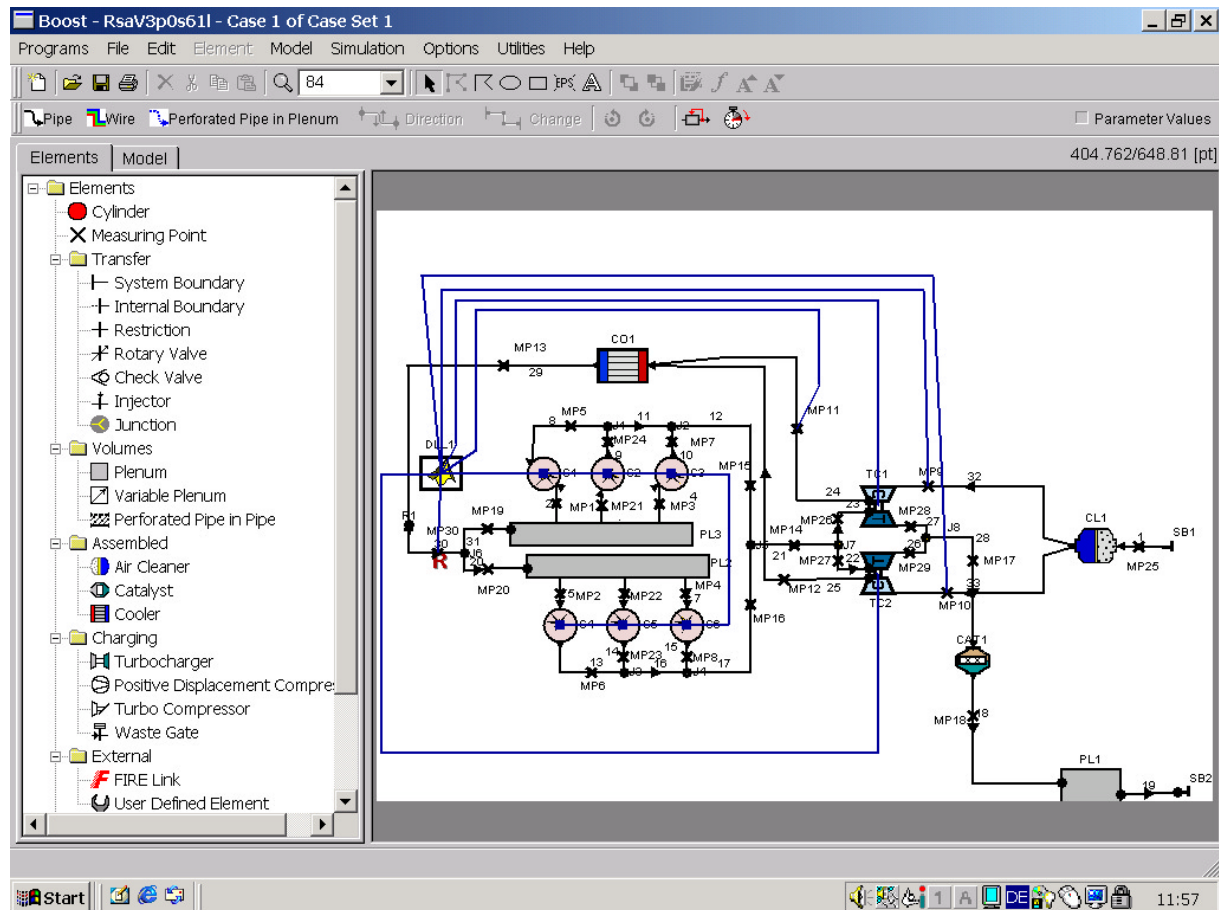


Abbildung 1: BOOST-Modell des Biturbo-Motors

Die VTG der Turbolader wird dabei sowohl zur Regelung des Ladedruckes, als auch für die bei dieser Luftsystemarchitektur notwendige Gleichstellungsregelung der Frischluftmassenströme verwendet. Die Notwendigkeit für diese Gleichstellungsregelung ergibt sich, wie in der Einleitung erwähnt, aus der Zusammenführung der Frischluftmassenströme auf der Hochdruckseite der Turboladerkompressoren, was vor allem bei ungleicher Turbolader-Effizienz aufgrund von Fertigungs- Ungenauigkeiten oder Abnutzungserscheinungen kritisch für ein funktionierendes Luftsystem ist. Dieses Architekturmerkmal ist aus dem BOOST-Modell des Motors (Abb. 1) ersichtlich. Die sich daraus ergebende Identität des Ladedruckes auf beiden Zylinderbänken ist vor allem bei der immer wichtiger werdende Reduzierung der Emissionswerte hilfreich. Das Modell des Motors wurde mit Hilfe einer von der AVL List GmbH entwickelten DLL in die Umgebung von

Matlab/Simulink eingebunden, welche ihrerseits als Basis für den Reglerentwurf fungiert. Dabei werden sowohl in das BOOST- als auch in das Simulink-Modell spezielle Funktionsblöcke eingefügt, welche als Dateninterface zwischen den beiden Programmen dienen. Die in der Abbildung sichtbaren blauen Linien symbolisieren die Datenpfade von und zu Simulink. Der in das Simulink-Modell eingebundene S-Function Block (Abb.2) kapselt eine DLL, welche ein von der BOOST-Oberfläche aus vorverarbeitetes Modell beim Start der Simulation aus einer Datei liest. Diese DLL übernimmt dann die eigentliche Simulation von Motor und Luftsystem in Abhängigkeit der Stellgrößen des in Simulink implementierten Reglers, wobei Flußgeschwindigkeiten, Drücke und Temperaturen auf den Gasdynamikgleichungen beruhen.

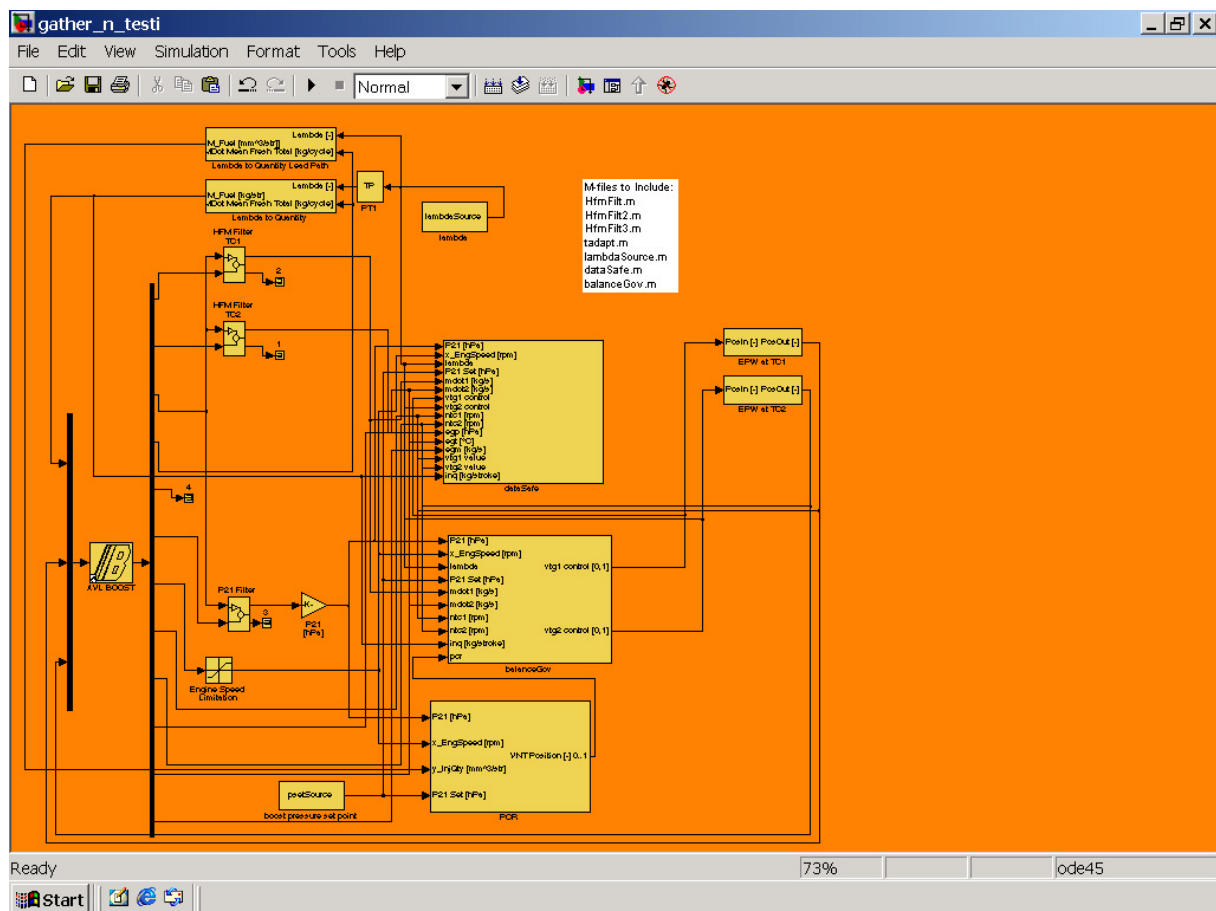


Abbildung 2: Das Simulink-Modell mit eingebundener BOOST-DLL

Nur bei den Durchflußbeiwerten werden dabei mehrdimensionale Effekte berücksichtigt, da es sich im Prinzip um eine eindimensionale Simulation handelt. Eine schematische Übersicht zur Einbettung einer BOOST-Simulation in die Matlab/Simulink Umgebung zur Erstellung einer gemischten Simulation ist in Abbildung 3 dargestellt. Diese Anbindung der BOOST-Simulation ermöglicht auch die Speicherung der relevanten Simulationsvariablen aus der Matlab/Simulink-Umgebung heraus mit Hilfe eines zu diesem Zweck Entwickelten S-Function Blockes.

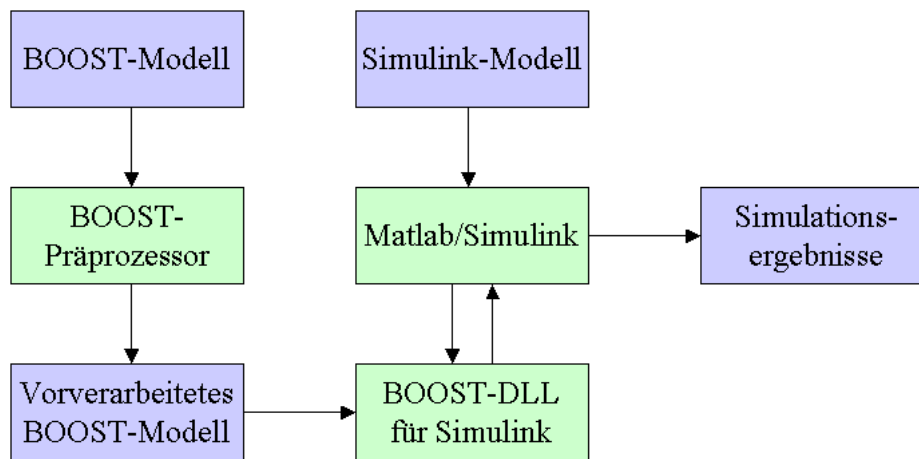


Abbildung 3: Einbettung einer BOOST-Simulation mittels DLL in Matlab/Simulink

3. Neuronaler Prädiktor

Mit Hilfe des entwickelten "Virtuellen Prüfstandes" wurden zunächst Daten aus mehreren Simulationsläufen für eine neuronale Black-Box Systemidentifikation gesammelt, welche näherungsweise die Zusammenhänge zwischen der Einspritzmenge, den VTG-Positionen und den Frischluftmassenströmen erfassen soll. Dazu wurden zunächst Daten bei identischen und unterschiedlichen Turboladereffizienzen gesammelt. Ein typischer Simulationslauf zur Erfassung von Daten für die Identifikation ist in Abb. 5 zu sehen. Hier ist die Effizienz des zweiten Turboladers gegenüber dem ersten um 30% erhöht.

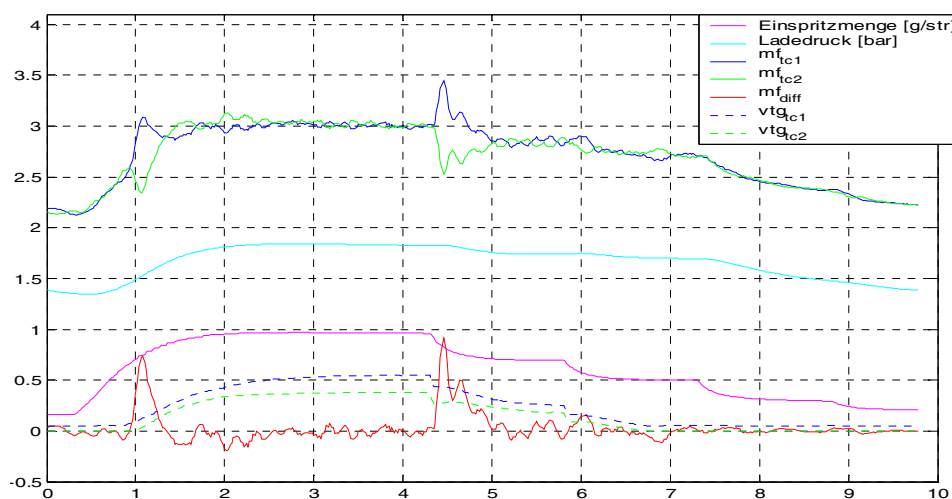


Abbildung 5: Typischer Simulationslauf zur Datenerfassung für die Identifikation

Als Prädiktor wurde ein voll vernetztes mehrlagiges Perzeptron mit einer versteckten Schicht aus 16 Knoten verwendet. Ein größeres Netz verbietet sich aufgrund der Laufzeitanforderungen an die Steuerung. Die Abbildungen 6 bis 8 zeigen die 'reale' und vorhergesagte Differenz zwischen den Turboladermassenströmen für 1-, 2- und 3-Schritt-Vorhersage bei 20ms Schrittweite.

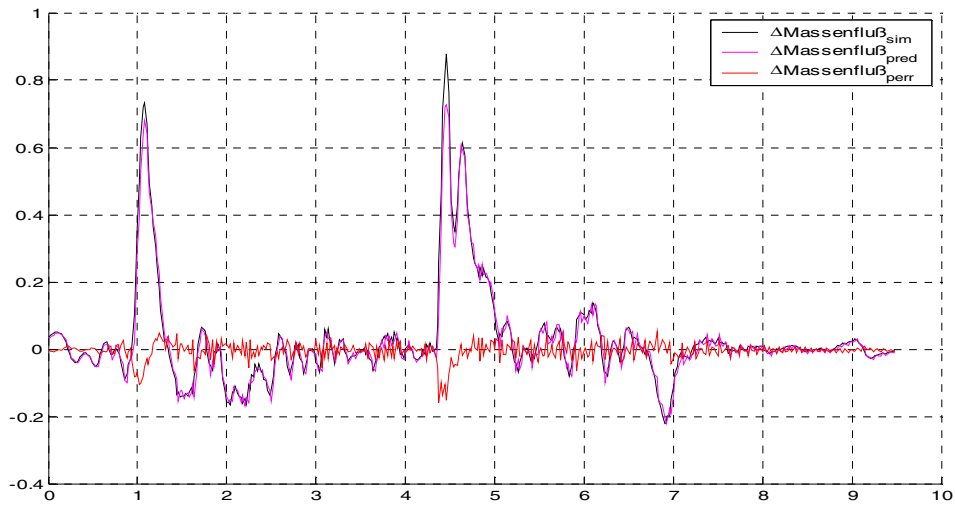


Abbildung 6: 'Reale' und vorhergesagte Massenstromdifferenz bei 1-Schritt-Vorhersage ($\Delta t=20\text{ms}$)

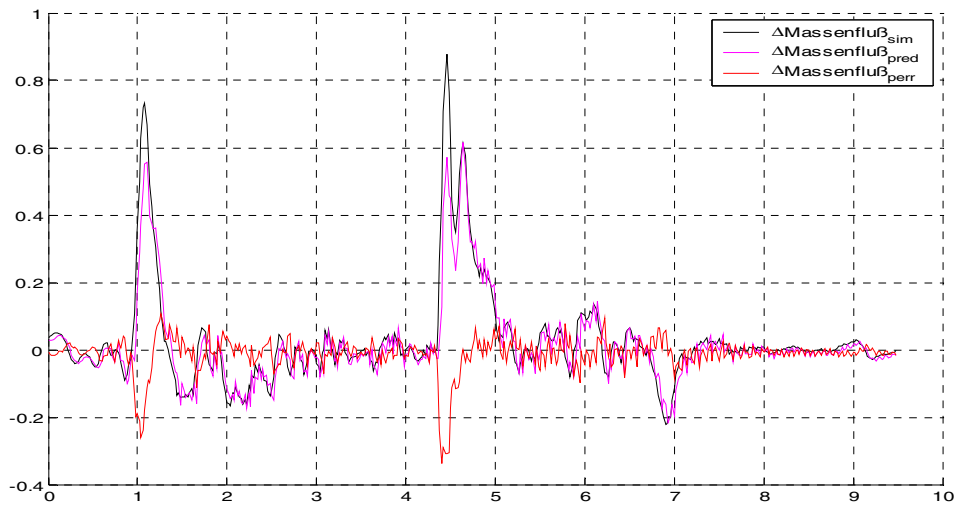


Abbildung 7: 'Reale' und vorhergesagte Massenstromdifferenz bei 2-Schritt-Vorhersage ($\Delta t=40\text{ms}$)

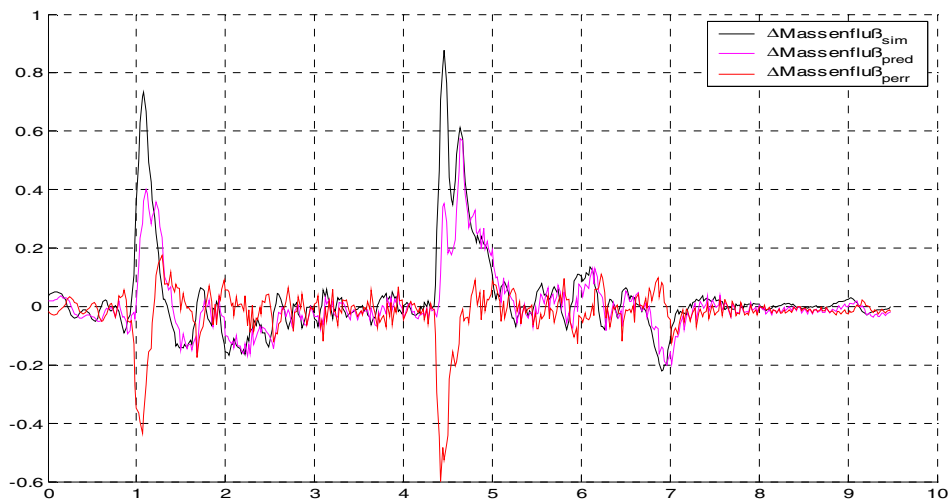


Abbildung 8: 'Reale' und vorhergesagte Massenstromdifferenz bei 3-Schritt-Vorhersage ($\Delta t=60\text{ms}$)

Die normalisierte mittlere Abweichung der Vorhersagen zum 'realen' (Simulations-) Wert beträgt 12%, 27% bzw. 47% für die 1-, 2- und 3-Schritt-Vorhersage. In einem experimentellen PI-Regler wurde der dem Proportionalregler zugeführte Fehlerterm aufgrund der Differenz zwischen Sollwert (bei Balanceregler immer gleich Null) und *vorhergesagtem* Massenstromdifferenz anstelle der augenblicklichen Massenstromdifferenz berechnet. Dieses einfache Regelschema ist dargestellt in der folgenden Abbildung 9.

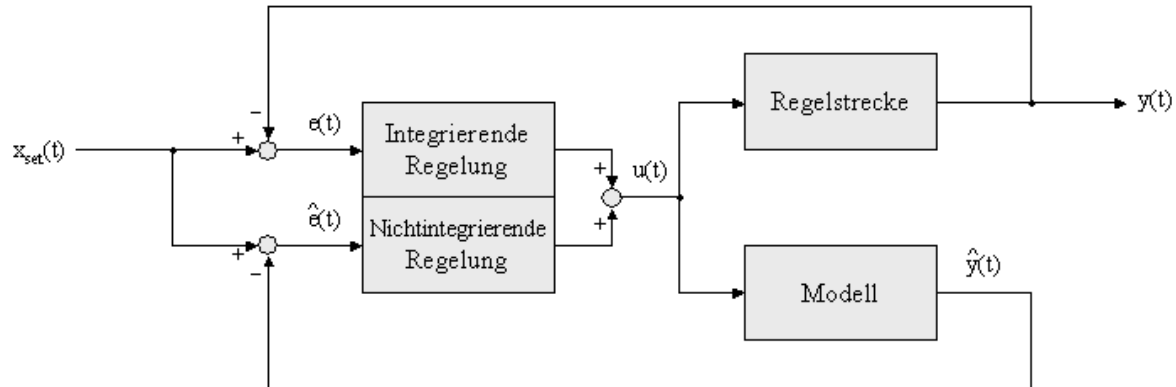


Abbildung 9: Ersetzung des Fehlerterms im P-Pfad des PI-Reglers durch vorhergesagten Fehler

Simulationsergebnisse (für einen Lambdasprung von 5 auf 1.15 bei einer konstanten Drehzahl von 1500 U/min und einer um 30% höheren Effizienz des zweiten Turboladers) bei PI-Regelung ohne bzw. mit Prädiktor finden sich unter Abbildung 10 und 11.

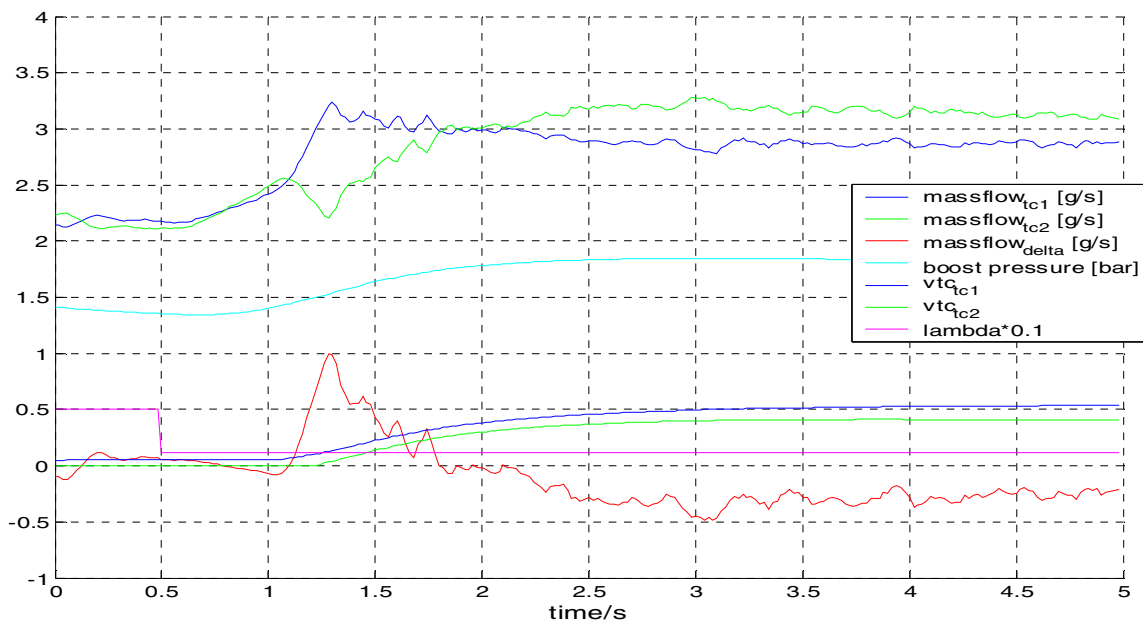


Abbildung 10: Lambdasprung von 5 auf 1.15 bei 1500 rpm, $\text{eff}_{\text{TC2}}=1.3 \cdot \text{eff}_{\text{TC1}}$, ohne Prädiktor

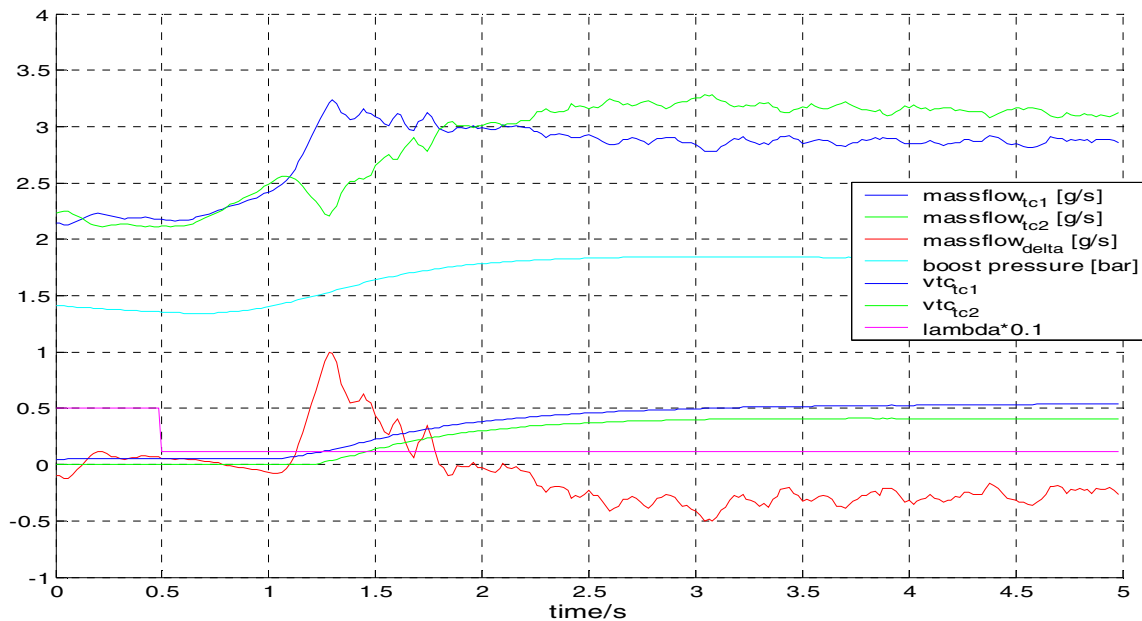


Abbildung 11: Lambdasprung von 5 auf 1.15 bei 1500 rpm, $\text{eff}_{\text{TC2}}=1.3 \cdot \text{eff}_{\text{TC1}}$, mit Prädiktor

Das Verhalten des PI-Reglers ist in beiden Fällen fast identisch, das Einfügen des Prädiktors in den proportionalen Regelpfad hat kaum Auswirkungen auf die Regelgüte.

Bewertung/Ausblick

Eine eindimensionale finite Elemente Simulation eines Biturbo-Motors wurde zur Generierung von Trainingsdaten für ein Künstliches Neuronales Netz zur dynamischen Vorhersage der Massenflußbalance zwischen beiden Turboladern verwendet. Es wurde erprobt, ob sich die Effizienz des Gleichstellungsregler für die Turbolader-Massenflüsse durch einfaches Einschleifen des KNN in den Proportionalpfad eines PI-Reglers verbessern läßt. Unabhängig vom Ergebnis des 'Experiments im Experiment' hat sich gezeigt, daß die Erprobung und vorläufige Bewertung neuer Regelansätze unter Ausnutzung des Konzeptes des 'virtuellen Prüfstandes' auch ohne teure Experimente am realen System möglich ist.

Referenzen

- [Münzenmay02] Münzenmay, Micha
 Implementation of Engine Cycle Simulation in the
 Function Development for Modern Electronic Diesel Control
 and its Application on Adaptive Balancing Control
 MPhil Report, 2002, Robert Bosch GmbH

Materialflusssimulation in der Flugzeugstrukturmontage

Eike Klemkow, Airbus Deutschland GmbH, eike.klemkow@airbus.com
Martin Falk Majohr, TU Dresden

Kurzfassung

Der Beitrag befasst sich mit einer Materialflusssimulation in der Flugzeugstrukturmontage. Diese Simulation wird hier als Optimierungstool zur Konzipierung einer neuen Fertigungsline eingesetzt.

Dabei steht die Ermittlung eines bestmöglichen Hallenlayouts unter Berücksichtigung der systemrelevanten Parameter wie Auftragseinstellung, Ressourceneinsatz und Produktionslogistik im Mittelpunkt. Es werden Wege und Möglichkeiten aufgezeigt, wie mit einem vertretbaren wirtschaftlichen Aufwand eine Materialflusssimulation aufgebaut wurde und in welchen Schritten eine Detaillierung des Modells und eine Erhärtung der Datenbasis erforderlich ist.

1 Einleitung

Charakteristisch für die Flugzeugstrukturmontage sind geringe Stückzahlen und Taktraten, ein hoher Anteil an Handarbeit sowie komplexe Bauelemente und Bauplätze. Signifikant sind dabei die extrem langen Durchlaufzeiten von über einem Jahr pro Flugzeug. Sie unterscheidet sich damit deutlich von den hohen Stückzahlen in der Automobilbranche und den hohen Taktfrequenzen in stark automatisierten Produktionsbereichen. Die komplexen Strukturen stellen besondere Anforderungen an die Planung und den Betrieb der Montagebauplätze sowie der Transporteinrichtungen.

Mit Hilfe der simulationsgestützten Planung wird im Airbuswerk Hamburg Finkenwerder eine neue Fertigungsline für die Strukturmontage entwickelt. Dabei werden gleichzeitig eine neue Fertigungsstruktur und neue Montagetechnologien für ein hinsichtlich des unternehmensinternen Workshares modifiziertes Produkt geplant. Da es sich hierbei um ein grundlegend neues Fertigungskonzept handelt, kann nur bedingt auf vorhandene Daten zurückgegriffen werden. Um den Planungszeitraum möglichst kurz zu halten, wird mit Hilfe von Simulationsansätzen versucht, ein genaues Abbild der zu erwartenden Auslastungen und des Prozessverhaltens zu geben, um die richtigen unternehmerischen Entscheidungen herleiten zu können. Dabei sollen bereits in der Planungsphase relativ belastbare Aussagen über Investitionskosten, Strategien für den Mitarbeiterinsatz, die optimale Einsteuerung der Bauteile und die zukünftige logistische Struktur getroffen werden.

2 Systembeschreibung

Die zu simulierenden Abläufe in der Großbauteilfertigung der Flugzeugstrukturmontage sind weniger komplex hinsichtlich der Abläufe (Anzahl der Stationen und Steuerungsmöglichkeiten), sondern zeichnen sich vielmehr durch hohe Variabilität in den einzelnen Fertigungsschritten aus. Die Workload in den einzelnen Arbeitsstationen variiert sehr stark bei den unterschiedlichen Modellen und Kunden, da sich die komplette Fertigung des Flugzeuges nach Kundenaufträgen richtet. Hinzu kommt, dass aufgrund der geringen Stückzahlen hohe Ansprüche an die Flexibilität der Fertigungseinrichtungen gestellt werden, um eine kostengünstige Produktion zu gewährleisten. In Bild 1 wird ein Überblick über die Variantenvielfalt der Sektionen der jeweiligen Flugzeugtypen, die durch eine einzige Fertigungslinie abzudecken sind, gegeben.

Sektionsbetrachtung:

Baumuster		Sektionen Rumpfvorderteil				Sektionen Rumpfhinterteil			
		13	14/14C	14A	14B	16	17	18	19
Widebody	A 310 PAX	8	9			4	14	8	9
	A 300-600 Frachter	8	15			10	14	8	9
	A 300-600 PAX	8	15			10	14	8	9
Long Range	A 330-200	8	14			4	14	8	9
	A 330-300	8	16	4		8	14	8	9
	A 340-200	8	16			4	14	8	9
	A 340-300	8	16	4		8	14	8	9
	A 340-500	8	6		14	10	14	8	9
	A 340-600	8	16		14	14	14	8	9

Bild 1: Bauteilvariation in der Strukturmontage

Die Prozessstruktur, die diesem Projekt zugrunde liegt, wird im Bild 2 dargestellt. Dabei handelt es sich um eine gradlinige Fließmontage, in der alle Sektionen die gleichen Bauplätze durchlaufen müssen. Innerhalb der einzelnen Bauplätze, in denen die Bauteile bis zu 60 Arbeitsstunden verbleiben, wird nach dem Werkstattprinzip gearbeitet, da hier immer gleiche Betriebsmittel zusammengefasst sind.

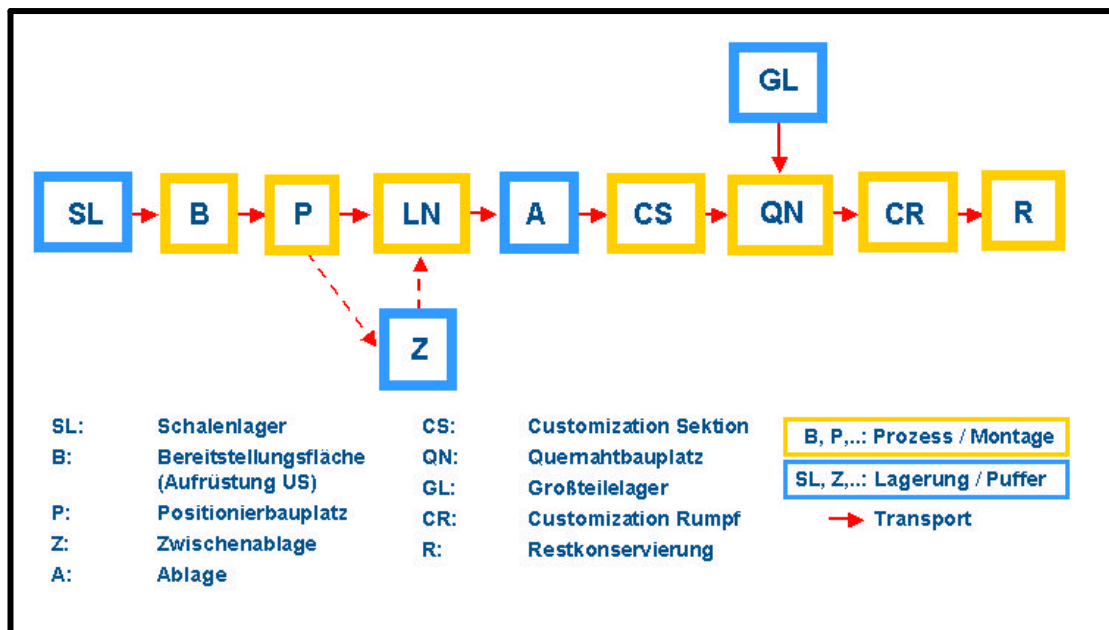


Bild 2: Prozessstruktur

3 Grundlagen der Modellierung

Für die Bildung des Modells wurden die wichtigsten Parameter definiert und festgelegt. Dabei handelt es sich vor allem um die relevanten Zeiten, zukünftige Kadenzen, eine Grundstruktur der Fertigung sowie spezielle Rahmenbedingungen der Fertigung. Teilweise wurde anfänglich nur mit groben Abschätzungen gearbeitet, die erst im Laufe des Projektes weiter untersetzt werden konnten. Der Detaillierungsgrad ist so definiert, dass nur die ressourcenbestimmenden Elemente in die Simulation aufgenommen werden. Dies sind im Speziellen alle Großkomponenten wie Teilschalen, Fußbodenroste und wichtige prozessbestimmende Fertigungshilfsmittel. Damit kann die Anzahl der in der Simulation verwendeten Elemente auf ein akzeptables Maß begrenzt werden, ohne einen wesentlichen Verlust an Genauigkeit im Modell zu erleiden. Die Durchlaufzeiten der einzelnen Bauteile für die verschiedenen Prozessschritte wurden auf Basis grob kalkulierter Auftragszeiten und einem statisch bestimmten Mitarbeiterereinsatz an der jeweiligen Arbeitsstation ermittelt. In Bild 3 ist exemplarisch die Durchlaufzeitkalkulation für eine spezifische Sektion abgebildet. Dabei wurden nur die wesentlichen Arbeitsschritte, die auf die Durchlaufzeit Einfluss haben, dargestellt und anhand der maximal am Arbeitsplatz einsetzbaren Anzahl von Mitarbeitern parallelisiert. Diese Kapazitätsbelastungsprofile existieren für alle Sektionen mit jeweils unterschiedlicher Anzahl von Mitarbeiter. Daraus sind dann entsprechende Funktionen zwischen der Mitarbeiteranzahl und der Durchlaufzeiten ableitbar.

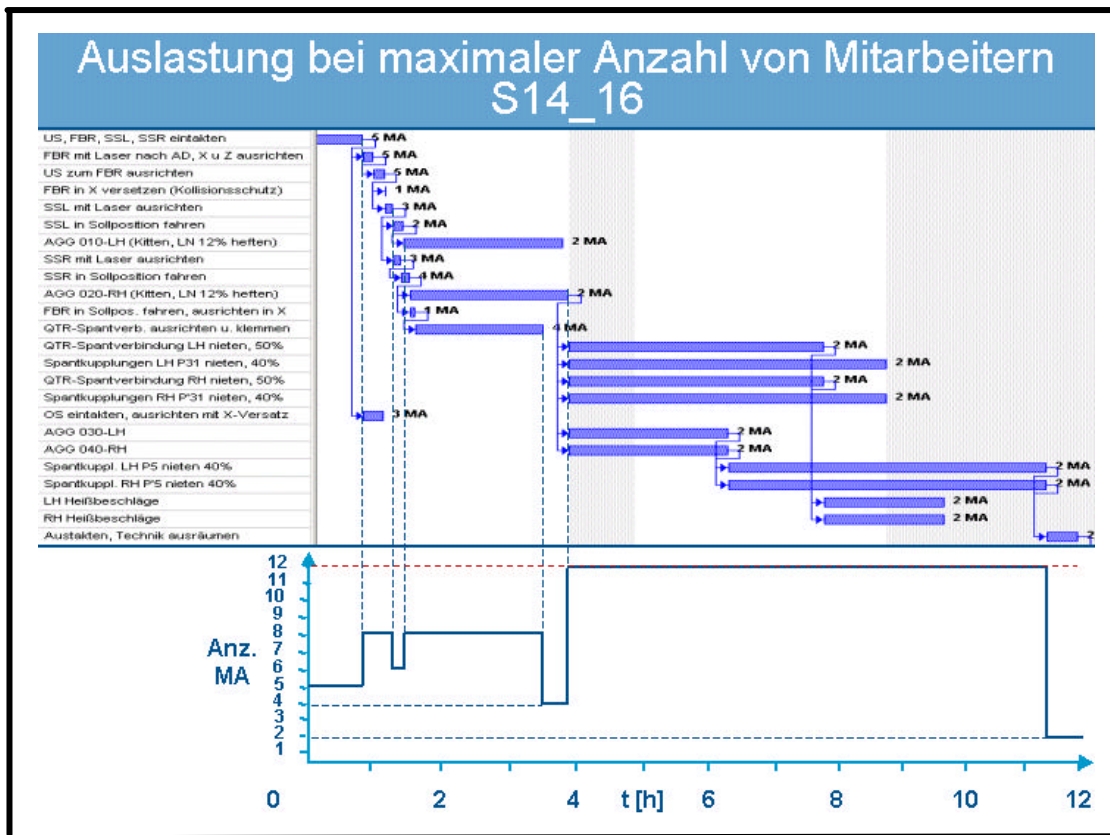


Bild 3: Prozessablauf und Mitarbeiterereinsatz am Beispiel einer Sektion mit 14 Spanten

Diese Funktionen können dann, kombiniert mit einem intelligenten Auftragseinsteu-
rungstool helfen, den kostenoptimalen Fluss durch die Fertigung zu ermitteln. Dieser ist
gegeben, wenn die Auslastung der Bauplätze, bei gleichzeitig minimaler Nutzung der
Ressourcen und einem weitgehend homogenen Fertigungsfluss maximal ist. Mit diesem
Simulationsmodell können die entscheidenden Systemparameter vorausbestimmt werden
und unter den einzelnen „Extremszenarien“ virtuell getestet werden. Aus den daraus ge-
wonnenen Erkenntnissen kann schon in einer sehr frühen Projektphase ein weitgehend
realistisches Fertigungslayout ermittelt werden. Anhand des erzeugten Layouts kann dann
eine detailliertere Betriebsmittelplanung und damit verbunden eine erste Investitionskosten-
abschätzung erfolgen. Dank der graphischen Animation der Simulation wird die Anschau-
lichkeit erhöht, so dass sie zum besseren Prozessverständnis beiträgt.

4 Planungsphasen

Für die Auslegung der Fertigung wird nach den in Bild 5 dargestellten Planungsphasen vorgegangen. Dadurch kann bereits auf Grundlage einer sehr unscharfen Datenbasis durch ein Layout eine Abschätzung der Investitionskosten erfolgen. Die folgende Auflistung zeigt, dass eine Vielzahl von Simulationsexperimenten notwendig ist, um den komplexen Zusammenhang zwischen einzelnen, das Layout bestimmenden, Komponenten zu ermitteln. Dabei ist der Prozess hinsichtlich der Auftragseinstellung, der Mitarbeiter-einteilung, der Pufferbestimmung sowie der Bauplatzverkettung zu optimieren. Da die Prozessstruktur in der Großbauteilfertigung sehr einfach ist, sind an dem aus der Planungsphase 1 resultierendem Layout und den zu erwartenden Stationsanzahlen keine wesentlichen Änderungen aus der Planungsphase 2 zu erwarten. Somit kann bereits in einem frühen Projektstadium mit der Planung der Immobilien begonnen werden und es liegen belastbare Daten für die Beantragung von Investitionsgeldern vor. Das Vorgehen nach der Planungsphase 1 ist zum gegenwärtigen Zeitpunkt abgeschlossen und die erwarteten Ergebnisse wurden erreicht.

Planungsphase 1 Grobe Daten	I <u>Statische Auslegung</u> II <u>Dynamische Simulation</u> 1. Auftragseinstellung der Sektionen bei einem bestimmten Flugzeugtyp 2. Auftragseinstellung der Flugzeugtypen 3. Festlegung der Pufferanzahl 4. Optimierung der Bauplatzanzahl 5. Ermittlung von Engpässen und der optimalen Bauplatzanzahl 6. Suchen der Systembelastungsgrenze III <u>Investitionskostenabschätzung</u>
Planungsphase 2 Feine Daten	IV <u>Mitarbeiter-einsatzkonzepte</u> V <u>Stückkostenabschätzung</u> VI <u>Ermittlung des Prozessverhaltens</u>

Bild 5: Planungsphasen für die Flugzeugstrukturmontage

Das aus der Planungsphase 1 abgeleitete Fabriklayout bietet die Grundlage für die Planungsphase 2, die zur Verifizierung von Mitarbeiter-einsatzkonzepten und zur weiteren Erhärtung der Kosten der zukünftigen Fertigung dient, sowie das genaue Prozessverhalten des Systems analysiert. Daraus sollen dann Ansätze für eine ereignisorientierte Prozesssteuerung generiert werden.

5 Zusammenfassung

Aufgrund der besonderen Struktur des Fertigungsprozesses in der Flugzeugstrukturmontage ist es möglich mit einer relativ unscharfen Datenbasis nach dem beschriebenen Planungsansatz schnell und mit einem vertretbaren wirtschaftlichem Aufwand mit Hilfe von nur einem Simulationsmodell eine belastbare Planung zu generieren. Dieses kann anschließend mit relativ geringem Modifikationsaufwand an vielen Stellen des Unternehmens Einsatz finden, da es eine Art „Wissensspeicher“ des gesamten Projektes darstellt. Die Anforderungen an die Simulation und die geforderte Funktionalität steigen mit Fortlauf der Planungen stark an. Hierfür ist es noch erforderlich geeignete Abstraktionen und Vereinfachungen zu definieren, um den wirtschaftlichen Aufwand gering zu halten.

6 Literatur

- [1] *Beese, D.*: Entwicklung eines Simulations-/Visualisierungsmoduls für Bauplatzuntersuchungen in der Flugzeugstrukturmontage. Report, Hochschule Wismar, 2002
- [2] *Streck, S.*: Simulation und Visualisierung in der Flugzeugstrukturmontage. Diplomarbeit, Hochschule Wismar, 2001
- [3] *Kremp, M.*: Materialflusssimulation in der Grossteilfertigung. Report, Hochschule Wismar/Airbus GmbH Deutschland, Hamburg, 2000

Untersuchung verschiedener Strategien zur simulationsbasierten Steuerung von Materialflusssystemen

M. Kremp, T. Pawletta, S. Pawletta, Hochschule Wismar
G. Colquhoun, Liverpool John Moores University

Kurzfassung

Dieser Beitrag untersucht die Umsetzung von operativen Prozesssteuerungen mit prozessgekoppelten Simulationsmodellen im Zusammenhang mit intelligenten Materialflussteuerungen. Ausgehend vom allgemeinen Konzept intelligenter Steuerungen wird die Erweiterung von Simulationskomponenten um eine Prozessschnittstelle aufgezeigt. Anschließend wird untersucht, wie aus der Steuerungstechnik bekannte operative Prozesssteuerungsansätze mit prozessgekoppelten Simulationsmodellen umgesetzt werden können. Es erfolgt eine Bewertung der unterschiedlichen Konzepte hinsichtlich ihrer Eignung zur Integration in intelligente Materialflussteuerungen.

1 Einleitung

Die Anforderungen an die Fertigungen im produzierenden Gewerbe steigen stetig. Die Wünsche nach einer kundenorientierten Produktion bis hin zur Losgröße Eins machen den Einsatz hochflexibler Fertigungseinrichtungen für stark und häufig veränderliche Anforderungsprofile notwendig. Für das Erreichen dieser Flexibilität muss auch die Steuerung eines Systems entsprechend leistungsfähig und variabel sein. Konventionelle Steuerungssysteme stoßen hier oftmals an ihre Grenzen.

An einer Lösung des Problems wird seit mehreren Jahren durch Weiterentwicklung von Prozesssteuerungen in Richtung regelungsbasierter Prozessführungen gearbeitet. Insbesondere im verfahrenstechnischen Bereich gibt es Lösungsansätze, speicherprogrammierbare Steuerungen (SPS) um Regelungskomponenten zu erweitern. Im Gegensatz zu vorrangig kontinuierlich ablaufenden verfahrenstechnischen Prozessen ist es für komplexe diskontinuierliche Produktionsprozesse nicht möglich, vorab ein vollständiges mathematisches oder heuristisches Reglergesetz zu definieren. Die dynamische Wirkung von Stelleingriffen muss experimentell zur Prozesslaufzeit ermittelt werden. Dies erfolgt heute zumeist offline durch dynamische Simulationsmodelle im Rahmen der dispositiven Produktionsplanung. Die Umsetzung der Steuerungsstrategien am Prozess erfolgt mittels SPS. Dieser Ansatz ist durch einen hohen Softwareaufwand gekennzeichnet, da Simulationsprogramme und SPS-Programme parallel entwickelt werden müssen. Die Reaktionsfähigkeit auf Prozessstörungen und auf sich ändernde Anforderungsprofile ist durch die Offline-Kopplung zwischen kurzfristiger Produktionsplanung und Prozesssteuerung begrenzt.

In [1] wird ein Prozessführungsansatz auf Basis eines prozessgekoppelten Simulators vorgestellt. Der prozessgekoppelte Simulator besteht aus drei Schichten. Die unterste Schicht bildet eine Prozessschnittstelle zur Online-Kopplung des

Simulators mit dem Prozess oder einer SPS. Die darüberliegende Schicht implementiert eine Menge operativer Steuerstrategien in Form von Realzeit-synchronisierten Simulationsmodellen. In der obersten Schicht werden eine Menge von dispositiven Simulationsmodellen für prädiktive Prozesssimulationen zur Verfügung gestellt. Ausgehend vom aktuellen Prozesszustand soll das Führungssystem durch prädiktive Prozesssimulationen vorausschauend Engpässe erkennen beziehungsweise bei aufgetretenen Störungen selbständig geeignete operative Steuerungsvarianten ermitteln.

Gegenstand dieses Beitrags ist die Realisierung operativer Prozesssteuerungen mit prozessgekoppelten Simulationsmodellen am Beispiel von Materialflusssystemen. Es werden unterschiedliche Ansätze untersucht und hinsichtlich ihrer Eignung zur Umsetzung eines intelligenten Prozessführungssystems bewertet. Zuvor werden das allgemeine Konzept simulationsbasierter Prozessführungen und die modulare Erweiterung von Simulationskomponenten um eine Prozessschnittstelle dargestellt.

2 Konzept einer intelligenten Materialflusssteuerung mittels prozessgekoppelter Simulation

Das Grundprinzip einer intelligenten Materialflusssteuerung, basierend auf dem regelungstechnischen Rückkopplungsprinzip ist in Abbildung 1 dargestellt. Ziel der intelligenten Steuerung ist nach [2] eine weitgehende Kompensation von internen und externen Störeinflüssen durch die Bestimmung geeigneter dispositiver Steuerungsstrategien. Störungen beeinflussen die momentane Systemlast, welche durch die Fertigungsaufträge und deren Arbeitsinhalte definiert wird. Sollgrößen ergeben sich aus den logistischen Zielgrößen wie beispielsweise Fertigungstermine und maximal zulässige Durchlaufzeiten. Stellgrößen sind zum Beispiel Einschleusungsreihenfolgen von Aufträgen sowie Zuordnungen von Maschinen- und Personalkapazitäten. Die Istwerte ergeben sich aus den aktuellen Zustandsgrößen des Produktionsprozesses, wie zum Beispiel den Pufferinhalten, Maschinenbelegungen und aktuellen Auftragszuständen.

Das intelligente dispositive Steuerungssystem besteht aus einem Prädiktions- und Bewertungsmodell. Beim Eintreten von Prozessgrenzzuständen oder bei Änderungen der logistischen Zielgrößen werden durch die dispositive Steuerung Prädiktionssimulationen zur Bestimmung neuer Steuerstrategien durchgeführt. Die Realisierung der Strategien am Prozess erfolgt durch das operative Steuerungssystem, welches über eine bidirektionale Online-Kopplung mit der übergeordneten dispositiven Ebene verbunden ist.

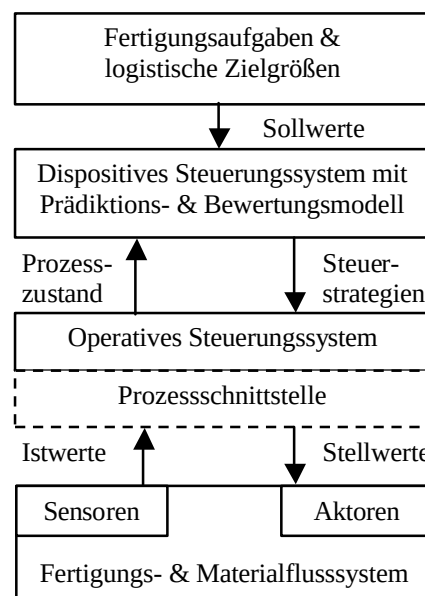


Bild 1: Prinzip einer intelligenten Materialflusssteuerung

3 Realisierung prozessgekoppelter Simulationskomponenten

Die Funktionalität von Simulationskomponenten moderner Materialflusssimulatoren orientiert sich an den formalen Strukturierungen der Materialflusslehre. Nach [4] wird zwischen *Verzweigungen*, *Zusammenführungen*, *Transportknoten* und *Transportübergangselementen* unterschieden. In bezug auf eine bestimmte Materialflusskomponente bilden vorgelagerte Elemente Quellen und nachfolgende Elemente Senken. Von jeder Quelle geht ein Förderstrom λ_Q aus und umgekehrt wird jede Senke mit einem Förderstrom λ_S belastet.

Abbildung 2 zeigt beispielhaft die materialflusstechnische Struktur eines allgemeinen Transportknotens nach [4]. Eintreffende und wartende Transporteinheiten dürfen die Komponente nach steuerungsbedingten Belegungsregeln in Abhängigkeit vom Zustand des Knotens sowie der zu beliefernden Senken betreten. Der interne Transport erfolgt je nach technischer Auslegung stetig, halbstetig oder unstetig.

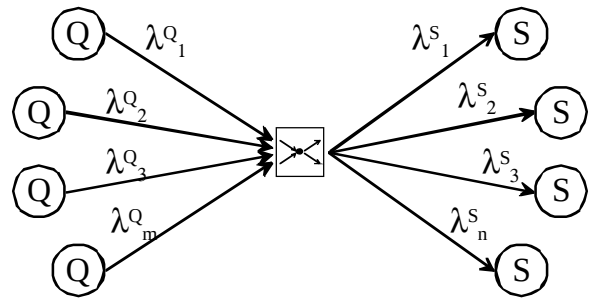


Bild 2: Beispiel eines (m,n) Transportknotens nach [4]

Soll ein Simulationsmodell nicht nur im Rahmen der Anlagen- und Prozessplanung eingesetzt werden, sondern auch zur Prozesssteuerung muß es über eine Prozessanbindung zur Auswertung von Sensorsignalen und zur Ansteuerung der Aktorik verfügen. In [1] wurde gezeigt, wie Simulationskomponenten von Planungssimulatoren systematisch um eine Prozessschnittstelle erweitert werden können. Die für Planungssimulationen entwickelte Funktionalität einer Simulationskomponente wird in einem Subsystem (*High Level System*) zusammengefaßt und um Ein-/Ausgangsschnittstellen zur Kommunikation mit der Prozessschnittstelle ergänzt. Die Prozessschnittstelle wird in einem separaten Subsystem (*Low Level System*) implementiert. Die für die Kommunikation der Subsysteme notwendige Umformung von Sensorsignalen in Simulationsergebnisse und umgekehrt von Simulationsergebnissen in Aktorsignale erfolgt im *Low Level System*.

Abbildung 3 zeigt beispielhaft die Struktur einer derartig erweiterten Simulationskomponente am Beispiel eines (2,2) Transportknotens. Die Belegung durch einen Auftrag wird der Komponente in Form eines Eingangsergebnisses mitgeteilt. Das Ereignis enthält Informationen zum Teiletyp und zur Werkstückträgerkennung. Die Bezeichnung des Teiletyps wird an das *High Level Subsystem* übergeben, während die Kennung des Werkstückträgers an das *Low Level Subsystem* gesendet wird. Beide Subsysteme aktualisieren daraufhin ihre Zustandsvariablen, wobei das *Low Level System* zusätzlich das entsprechende Sensorsignal überprüft. Die Ausschleusungsrichtung wird durch den Eingangsport *IN_control* übergeben. Das Ausfahren eines Werkstückträgers aus dem Transportknoten wird vom *Low Level Subsystem* überwacht. Bei Eintreffen des entsprechenden Sensorsignals sendet es über die *C_finished_move_out* Kopplung ein Ereignis an das *High Level Subsystem*. Beide

Subsysteme aktualisieren ihre Zustandsvariablen und es wird ein Ausgangsereignis mit Informationen zum Teiletyp sowie der Werkstückträgerkennung generiert.

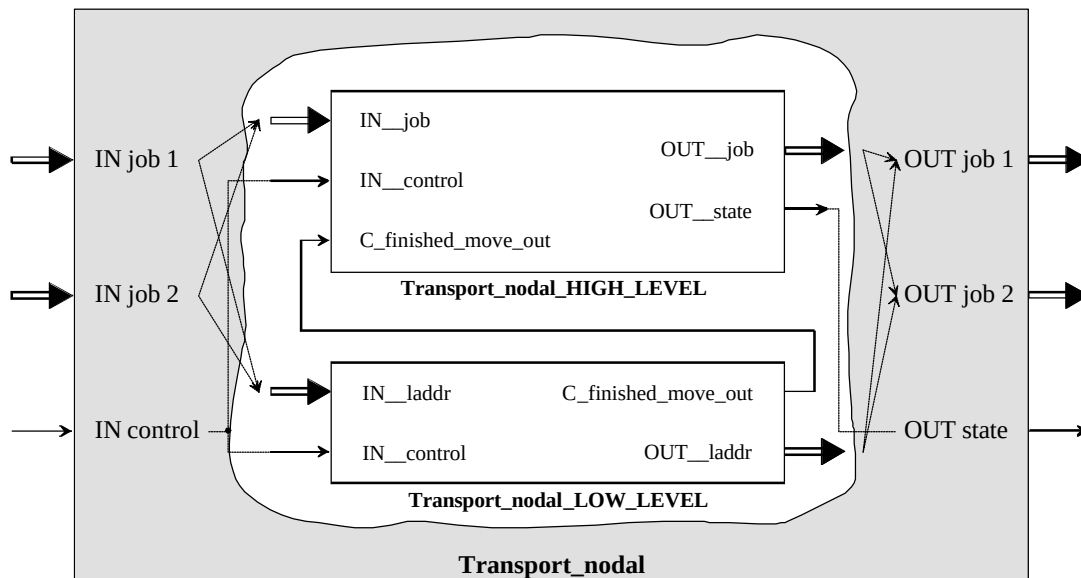


Bild 3: Simulationskomponente eines (2,2) Transportknotens

4 Realisierung prozessgekoppelter Simulationsmodelle zur operativen Prozesssteuerung

Auf der Basis der im vorangegangenen Abschnitt konzipierten Simulationskomponenten soll nachfolgend die Umsetzung von operativen Prozesssteuerungen mit prozessgekoppelten Simulationsmodellen untersucht werden. Es wird von der allgemeinen Klassifikation in verbindungsorientierte, zentralisierte und verteilt-hierarchische Steuerungen gemäß [5,6] ausgegangen.

Voraussetzung einer simulationsbasierten Prozesssteuerung ist eine Realzeit-Synchronisation des Simulationsmodells mit dem Prozess. Die Aspekte der Zeitsynchronisation sind in [1] ausführlich dargestellt und werden für die weiteren Betrachtungen als gegeben vorausgesetzt.

4.1 Verbindungsorientiertes Steuerungsmodell

Verbindungsorientierte Steuerungen basieren auf direkten booleschen Informationsbeziehungen zwischen den Anlagenelementen, die im Steuerungsmodell durch die Simulationskomponenten repräsentiert werden. Abbildung 4 zeigt ausschnittsweise die Realisierung einer simulationsbasierten operativen Steuerung nach diesem Prinzip. Der dargestellte Modellteil enthält einen Server mit einem nachgelagerten Puffer. Fördereinheiten dürfen den Server nur verlassen wenn ausreichend Platz im Puffer vorhanden ist. Die Simulationskomponente des Puffers stellt über den Ausgangsport *OUT_state* die aktuelle Pufferbelegung zur Verfügung. Durch die angeschlossenen Logikblöcke wird geprüft, ob der Puffer freie Kapazitäten besitzt und

es wird ein Steuerungssignal für die Simulationskomponente des Servers (Eingangsport *IN_control*) generiert. Ist eine Bearbeitung im Server beendet, prüft dieser das Steuerungssignal und entscheidet über die Ausschleusung der Fördereinheit. Ähnlich erfolgt die Ausschleusung von Fördereinheiten aus dem Puffer.

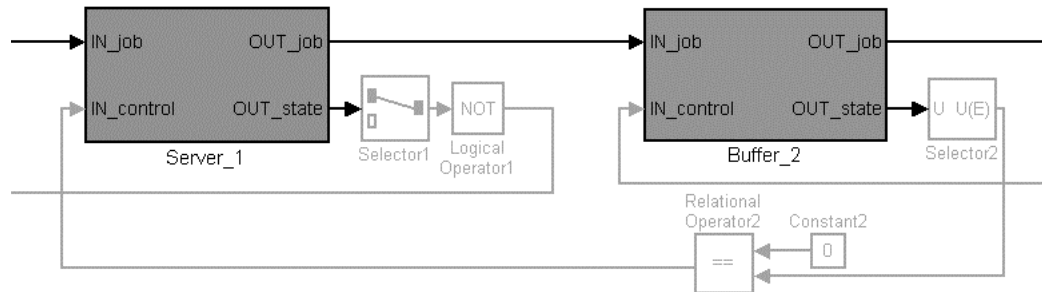


Bild 4: Ausschnitt eines Simulationsmodells mit verbindungsorientierter Steuerung

Das aufgezeigte Beispiel enthält ausschließlich steuerungstechnisch einfache Materialflusselemente. Die Integration von Verzweigungen, Zusammenführungen oder Transportknoten verlangt wesentlich umfangreichere logische Verknüpfungen. Damit führt bereits die Abbildung einfacher Steuerstrategien oft zu komplexen Modellen. Bei der Umsetzung variabler und alternativer Steuerungsvarianten nimmt die Modellkomplexität exponentiell zu. Eine definierte Schnittstelle zur übergeordneten dispositiven Steuerung existiert nicht.

4.2 Zentralisiertes Steuerungsmodell

Beim zentralisierten Steuerungsansatz ist die gesamte Steuerungslogik in einem separaten Modul integriert, das intern modular-hierarchisch aufgebaut sein kann. Der Informationsaustausch zwischen der Steuerungskomponente und den Materialflusskomponenten kann im prozessgekoppelten Simulationsmodell über globale Variablen oder Verbindungsvariablen, die Signalleitungen entsprechen, erfolgen. Abbildung 5 zeigt einen Ausschnitt aus einem beispielhaften Simulationsmodell, in welchem die Kommunikation mit der zentralen Steuerungskomponente über Verbindungsvariablen erfolgt.

Die Kommunikation mit der übergeordneten dispositiven Ebene und die Abbildung flexibler operativer Steuerungsstrategien ist aufgrund der Zentralisierung in einem Modul problemlos umsetzbar. Die strenge Trennung von Materialfluss und Steuerung birgt aber auch Nachteile, insbesondere bei der Hierarchisierung von Modellteilen und bei Modelländerungen. Außerdem nimmt die Komplexität der Steuerungskomponente bei großen und variabel steuerbaren Modellen schnell zu.

4.3 Verteiltes hierarchisches Steuerungsmodell

Bei komplexen Systemen ist es sinnvoll Komponenten hierarchisch zusammenzufassen. Um hierarchisierte Komponenten effektiv warten zu können, sollten lokal begrenzte Steuerungsoptionen mit in die hierarchisierten Komponenten integriert

werden. Daraus folgt ein verteiltes Steuerungskonzept. In der Steuerungstechnik wird bei verteilten Steuerungen gemäß [5,6] zwischen einer Leitebene, Gruppensteuerungen und Einzelsteuerungen unterschieden.

Das Simulationsmodell in Abbildung 5 enthält die hierarchische Komponente *Serversystem_1*. Abbildung 6 zeigt eine mögliche innere Struktur der Komponente *Serversystem_1* nach dem verteilten Steuerungsansatz, bestehend aus drei Servern, einem Verteilungs- und Zusammenführungselement sowie einer lokalen Gruppensteuerungskomponente. Die Gruppensteuerung kommuniziert mit der globalen Steuerung (Leitebene) und meldet dieser zum Beispiel das Ausschleusen von Fördereinheiten aus dem Serversystem. Die Auswahl eines Servers bei Eintritt einer Fördereinheit wird autonom durch die Gruppensteuerung getroffen.

Der Vorteil des verteilten Steuerungsansatzes gegenüber dem zentralisierten Ansatz liegt bei hierarchischen Modellen in einer besseren Strukturierung bei Erhaltung der Flexibilität. Durch die Gruppensteuerungen können gezielt alternative operative Steuerungsstrategien für abgegrenzte Systembereiche realisiert werden. Die Schnittstelle zur übergeordneten dispositiven Ebene bildet der globale Steuerungsmodul der operativen Steuerung.

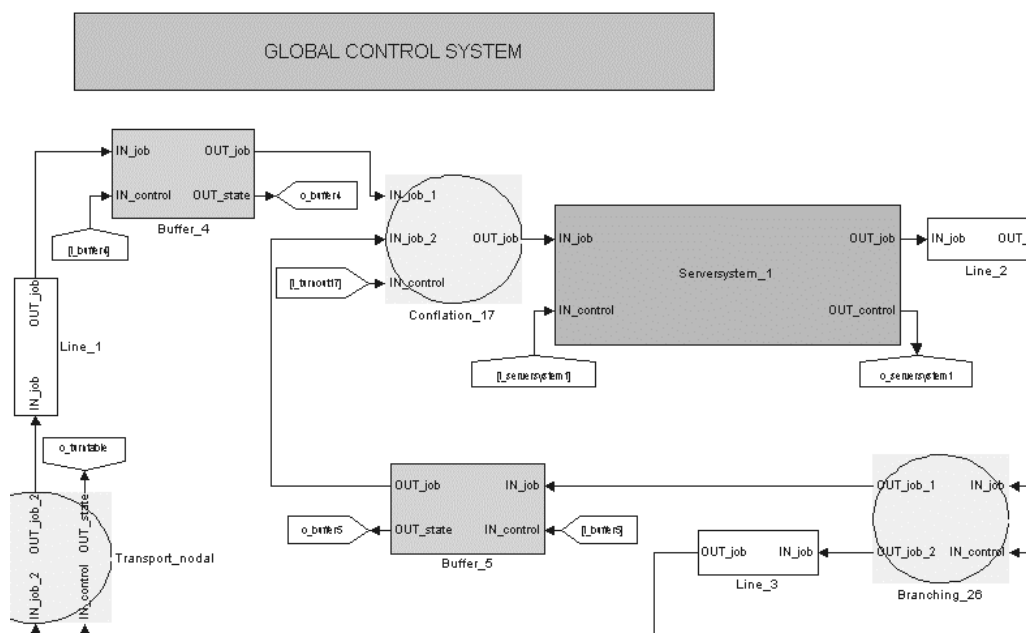


Bild 5: Ausschnitt eines Simulationsmodells mit zentraler Steuerung

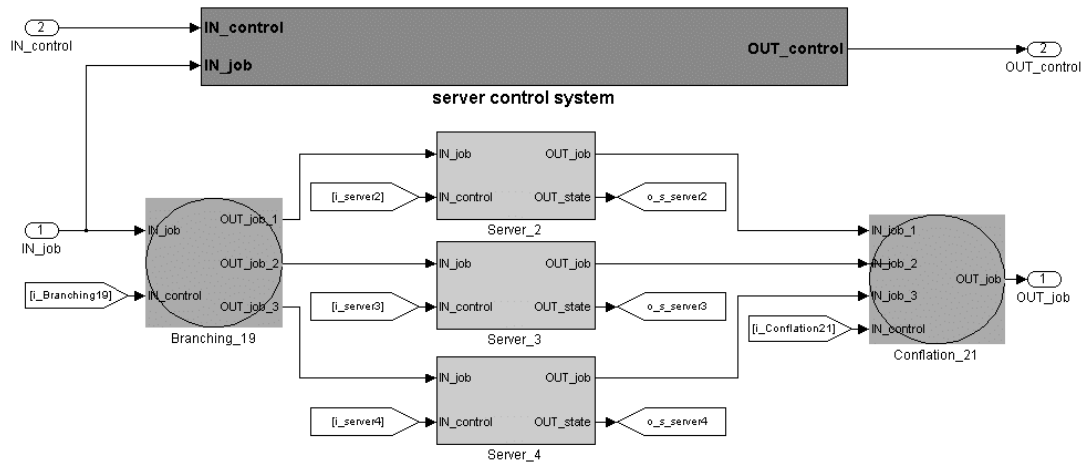


Bild 6: Struktur einer Simulationskomponente mit lokaler Gruppensteuerung

5 Zusammenfassung

Die im Beitrag aufgezeigte softwaretechnische Erweiterung von Simulationskomponenten um eine Prozessschnittstelle und die diskutierten Ansätze zur Realisierung von operativen Prozesssteuerungen mit prozessgekoppelten Simulationsmodellen wurden am Beispiel eines experimentellen Materialflusssystems untersucht.

Hinsichtlich der Strukturierung der Simulationskomponenten zeigte sich, dass diese für Planungssimulationen und mit der Erweiterung um eine Prozessschnittstelle direkt zur Prozesssteuerung eingesetzt werden können.

Bei den untersuchten Steuerungsvarianten ist erwartungsgemäß einzuschätzen, dass prozessgekoppelte Simulationsmodelle nach dem verbindungsorientierten Steuerungsansatz nur für relativ starre operative Steuerungen geeignet sind. Für einfach aufgebaute Materialflusssysteme sind prozessgekoppelte Simulationsmodelle nach dem zentralisierten Steuerungsansatz am besten geeignet, um intelligente Materialflussteuerungen umzusetzen. Bei komplexen Materialflusssystemen sollten prozessgekoppelte Simulationsmodelle nach dem verteilt-hierarchischen Steuerungsansatz aufgebaut werden, da nur dieser eine sinnvolle Modellhierarchisierung unterstützt.

6 Literatur

- [1] T. Pawletta, S. Pawletta, M. Kremp, L. Mohr, W. Drewelow: Intelligente Steuerung von Materialflusssystemen mit prozessgekoppelter Simulation, In D.Tavangarian, editor, Frontiers in Simulation - Simulationstechnik 16. Symposium Rostock, pages 203-208, SCS European Publishing House, Ghent 2002
- [2] C. Martin: Verfahren und Systeme zur simulationsbasierten Produktionsregelung bei komplexen Produktionsstrukturen, Fortschritte der Simulationstechnik, Bd.9, Vieweg Verlag, 1994, S. 591-596
- [3] Z. Marinsek: Der Einsatz von speicherprogrammierbaren Steuerungen in der Prozessregelung. <http://www.mitsubishi-automation.de/loesungen/art1.htm>, 2002
- [4] T. Gudehus: Transportsysteme für leichtes Stückgut, Materialfluss im Betrieb, VDI-Verlag, 1977
- [5] DIN 19226: Regelungstechnik und Steuerungstechnik, DIN Norm, Deutsches Institut für Normung e.V., Berlin, Februar 1994
- [6] G. Strohrmann: Automatisierungstechnik 1, Grundlagen, analoge und digitale Prozessleittechnik, R. Oldenbourg Verlag, München, Wien 1998

Modellbasierte Regelung in der Anästhesie

O. Simanski, R. Kähler, B.P. Lampe Universität Rostock *

B. Pohl, R. Hofmockel Universität Rostock †

Zusammenfassung

Motiviert durch die Entwicklung immer kürzer wirkender Muskelrelaxantien gehen die Anästhesisten zunehmend von einer Bolusapplikation der Medikamente zu einer kontinuierlichen Applikation über. Nachfolgend wird ein modellbasierter Regler für die kontinuierliche Medikamentengabe vorgestellt. Es werden die Entwurfsansätze und die Implementation beschrieben.

1 Einleitung und Motivation

Die Narkose ist als reversibler Bewusstseinsverlust, der mit Schmerzfreiheit, Muskelentspannung und Reflexdämpfung einhergeht, definiert. Bei modernen Kombinationsnarkosen werden die Teilkomponenten der Narkose, die Hypnose, die Analgesie und die Muskelrelaxation durch Medikamente wie Hypnotika, Analgetika und Relaxantien gezielt beeinflusst.

Während durch die ausreichende Hypnose und Analgesierung sichergestellt werden soll, dass der Patient nicht intraoperativ aufwacht oder Schmerzen empfindet, gewährleistet eine ausreichende Muskelrelaxation optimale Bedingungen für den Operateur.

Die Muskelrelaxation kann mit Hilfe eines neuromuskulären Monitorings messtechnisch erfasst werden. In der klinischen Routine wird der Grad der Muskelrelaxation oftmals visuell oder taktil¹ erfasst, das heißt, es werden die Reaktionen der innervierten² Muskeln durch Beobachtung oder Berührung erfasst. Grundlage für die Entwicklung eines Regelungssystems ist jedoch die quantitative Erfassung. Dazu stehen verschiedene, nachfolgend kurz dargelegte Verfahren zur Verfügung. Als Mechanomyographie (MMG) wird ein Verfahren bezeichnet, das isometrisch die Kraft des Muskels misst, der sich infolge der Nervenstimulation kontrahiert. Als Muskel wird dabei standardmäßig der Musculus adductor pollicis am Daumen, nach Stimulation des Nervus ulnaris, der sich am Unterarm befindet, verwendet [Hof97]. Ebenfalls ein mechanisches Verfahren ist die Acceleromyographie (AMG), sie misst unter Nutzung von Beschleunigungssensoren die Beschleunigung z.B. des Daumens. Das dritte und für den konkreten Anwendungsfall eingesetzte Messverfahren ist die Elektromyographie (EMG). Bei ihr werden mittels Edelstahlfächenelektroden Muskelaktionspotentiale, die vor Muskelkontraktion entstehen, an der

*Institut für Automatisierungstechnik, D-18051 Rostock, olaf.simanski@technik.uni-rostock.de

†Klinik und Poliklinik für Anästhesiologie und Intensivtherapie

¹den Tastsinn betreffend

²mit Nerven oder Nervenreizen versorgen, anregen

Hautoberfläche registriert.

Die mittels EMG erfasste neuromuskuläre Blockade bildet somit die Regelgröße und die applizierte Menge an Muskelrelaxans Mivacurium die Stellgröße. Der Mensch und seine Muskelrelaxation als Folge der Mivacuriumapplikation verkörpert die Regelstrecke.

2 Modellierung der neuromuskulären Blockade

Für eine Erprobung der zu entwerfenden Regelungsstrategien sind Simulationen vor dem Einsatz am Patienten unumgänglich. Ausgehend von der Pharmakologie der Relaxantien können mathematische Beschreibungen für die Pharmakokinetik und Pharmakodynamik entwickelt werden. Die Pharmakokinetik beschreibt die Einflüsse des Organismus auf das Pharmakon, d.h. Resorption, Verteilung und Elimination. Im Unterschied dazu werden mit der Pharmakodynamik die Einflüsse des Pharmakons auf den Organismus dargestellt. Die Beschreibung der Pharmakokinetik erfolgt dabei durch Kompartiment-Modelle. Der menschliche Körper wird dazu in relevante Körperkompartimente unterteilt. Die Modelle beschreiben Verteilung und Abbau innerhalb sowie den Austausch des Relaxans zwischen den Kompartimenten. Abbildung 1 zeigt das 2+1-Kompartiment-Modell für die Wirkung des Medikamentes Mivacurium auf den Grad der neuromuskulären Blockade.

Ausgehend von einem Zwei-Kompartiment-Modell werden die Medikamentenkonzen-

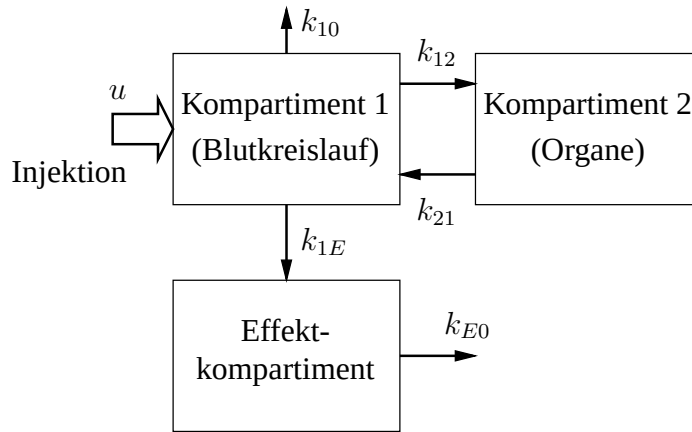


Abbildung 1: Graf. Darstellung des Zwei-Kompartimentmodells + Effekt-Kompartiment (2+1-Kompartiment-Modell)

tration im i -ten Kompartiment zur Zeit t durch x_i , die Änderung der Relaxanskonzentration durch $\dot{x}_i$ und die Relaxansgabe durch u dargestellt [ML98, Sim02]. Das Verhalten kann durch die Differentialgleichungen

$$\begin{aligned}\dot{x}_1 &= -(k_{10} + k_{12})x_1 + k_{21}x_2 + u \\ \dot{x}_2 &= k_{12}x_1 - k_{21}x_2\end{aligned}\tag{1}$$

beschrieben werden. k_{10} kennzeichnet die Konstante für die Eliminationsrate und k_{12} , k_{21} die Übertragungskonstanten zwischen den Kompartimenten. Das zentrale Kompartiment

wird durch den Index 1 und das periphere Kompartiment durch den Index 2 gekennzeichnet. Aus den Differentialgleichungen ergibt sich dann mit

$$G(s) = \frac{X_1(s)}{U(s)} = \frac{K(1 + sT_D)}{(1 + sT_1)(1 + sT_2)} \quad (2)$$

eine Übertragungsfunktion, die in der Verstärkung und den Zeitkonstanten auf das jeweilige Relaxans angepasst wird.

Dieses Modell zeigt für einige Relaxantien ein ungenügendes Verhalten. Daher wurde ein zusätzliches hypothetisches Effekt-Kompartiment

$$\dot{x}_E = k_{1E}x_1 - k_{E0}x_E \quad (3)$$

eingeführt. Bei dem ähnlich wie in (1) k_{E0} die Eliminationskonstante, k_{1E} die Übertragungskonstante und der Index E das Effekt-Kompartiment charakterisieren. Das hypothetische Effekt-Kompartiment erweitert die Übertragungsfunktion um eine weitere Verzögerung zu

$$G_s(s) = \frac{X_E(s)}{U(s)} = \frac{K(1 + sT_D)}{(1 + sT_1)(1 + sT_2)(1 + sT_3)}. \quad (4)$$

Das Modell wird um eine Ersatztotzeit ergänzt, in die die Transportdauer des Relaxans von der Injektionsstelle bis zum Wirkungsort eingeht. Es ergibt sich dann

$$G_{ST}(s) = \frac{X_E(s)}{U(s)} = \frac{e^{-\tau s} K(1 + sT_D)}{(1 + sT_1)(1 + sT_2)(1 + sT_3)}. \quad (5)$$

Zur Beschreibung der Sättigungseffekte wird die Hill-Gleichung verwendet

$$E_{eff} = \frac{E_{max}x_E^\alpha}{x_E^\alpha + x_E(50)^\alpha}, \quad (6)$$

die die Pharmakodynamik beschreibt. Die Medikamentenkonzentration wird mit x_E bezeichnet und $x_E(50)$ beschreibt die Relaxanskonzentration bei 50% neuromuskulärer Blockade, α ist der Hill-Koeffizient. Das Blockschaltbild des Gesamtmodells ist in Abbildung 2 dargestellt.

Das ermittelte Modell (5), (6) wurde unter Verwendung aufgenommener Identifikations-

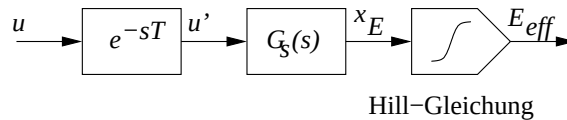


Abbildung 2: Blockschaltbild des theoretischen Kompartimentmodells

datensätze parametrisiert.

Eine Parametrisierung mittels Minimierung der Verlustfunktion (7)

$$V_N(\theta) = \frac{1}{N} \sum_{k=1}^N \varepsilon^2(k, \theta) \quad (7)$$

mit einem θ von

$$\theta = (K, T_1, T_2, T_3, T_D, x_E(50), \alpha), \quad (8)$$

und k als Messzeitpunkt sowie ε als Fehler zwischen Messung und Vorhersage, führte zu dem in Abbildung 3 gezeigten Ergebnis. Zur Parameterschätzung wurde die Nelder-Mead Simplex Methode genutzt. Der Modellfehler beträgt $RMSE = 2,88 \%$ neuromuskuläre Blockade für die folgenden ermittelten und neben der Abbildung 3 dargestellten Modellparameter einer Patientin.

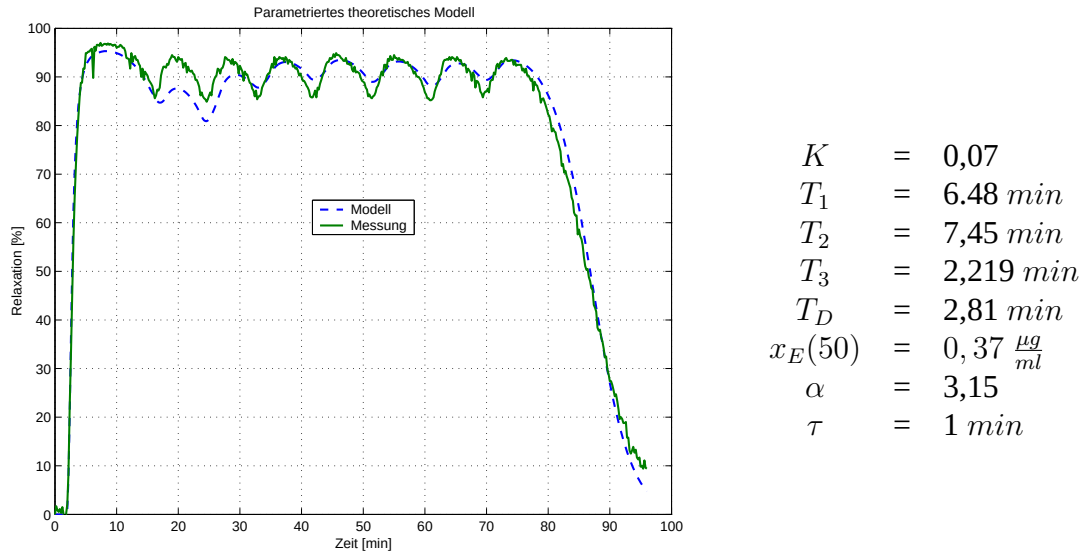


Abbildung 3: Simulation der Mivacuriumwirkung, (Beispiel)

3 Regelung der neuromuskulären Blockade mittels GPC

Das in Abschnitt 2 eingeführte nichtlineare Modell wurde für die Simulationen im Vorfeld des klinischen Einsatzes genutzt.

Der Prozess der neuromuskulären Blockade unterliegt sehr starken individuellen Veränderungen. Aus diesem Grund wurde ein Regler gesucht, der robust gegenüber unbekannter und variabler Totzeit sowie veränderlichen Modellparametern ist. Zusätzlich soll die Regelung in der Lage sein, sich an das mögliche veränderte Patientenverhalten zu adaptieren. Mit dem Generalized Predictive Controller (GPC) von Clarke [CMT87] konnten die gestellten Anforderungen zufriedenstellend gelöst werden.

Der GPC ist ein linearer Regler. Damit dieser an dem nichtlinearen Prozess der neuromuskulären Blockade eingesetzt werden kann, wurden lineare Bedingungen um den gewünschten Arbeitspunkt angenommen. Der lineare Arbeitsbereich erstreckte sich von 88 – 100% neuromuskulärer Blockade. Als Sollwert wurde eine 90%ige neuromuskuläre Blockade gewählt. Dem GPC, dessen Arbeitsweise in Abbildung 5 dargestellt ist, werden durch ein lineares ARX-Modell, die aktuellen Patienteneigenschaften übergeben. Durch

die Implementation der rekursiven ARX-Modellbildung, die mittels der rekursiven Methode der kleinsten Quadrate erfolgte, wurde der GPC zu einem adaptiven Regler erweitert.

Ein Modell $G_{online}(q^{-1})$ 3.Ordnung modelliert das Verhalten des Patienten um den Arbeitspunkt. Zusätzliche 3 $\hat{b}$ -Koeffizienten ermöglichen geringfügige Variationen in der Totzeit (9) [Lin94].

$$G_{online}(q^{-1}) = q^{-5} \frac{b_1 + b_2 q^{-1} + b_3 q^{-2} + b_4 q^{-3} + b_5 q^{-4} + b_6 q^{-5}}{1 + a_1 q^{-1} + a_2 q^{-2} + a_3 q^{-3}} \quad (9)$$

In der Abbildung 4 sind die online bei einer Regelung ermittelten Pol- und Nullstellen in der z-Ebene dargestellt. Durch die kleine Abtastzeit von $T_A = 12\text{ s}$ kommt es zu dem numerischen Problem, dass die eine Polstelle gegen den Wert 1 strebt. Außerdem schwanken die $\hat{b}$ -Parameter stark und es wird ein nichtminimalphasiges Modell identifiziert. Der

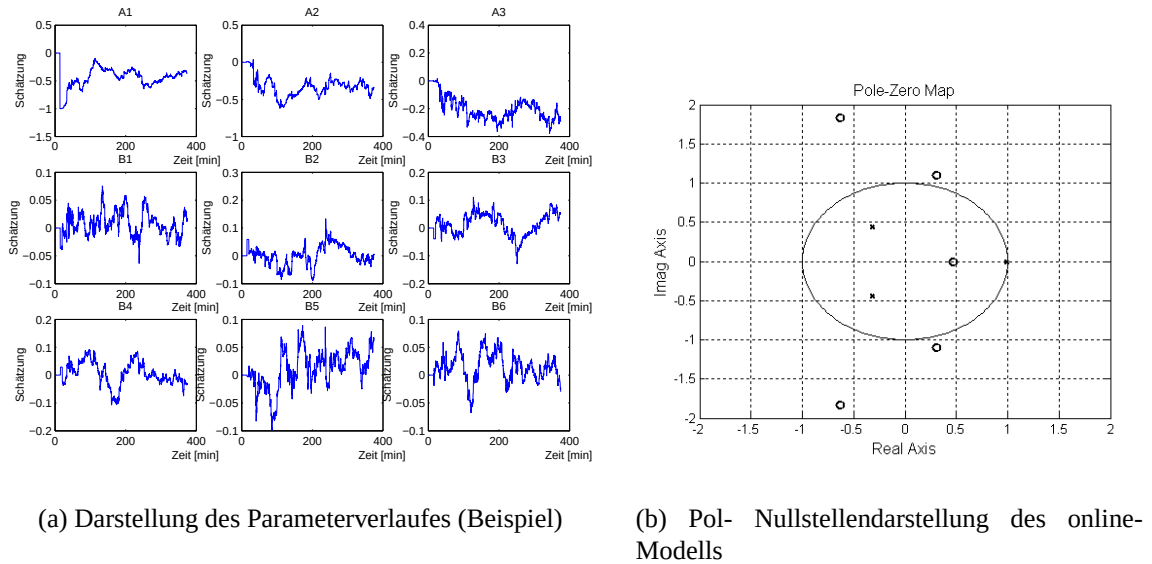


Abbildung 4: Beispieldatensatz und Pol- Nullstellendarstellung des linearen online-Modells

GPC ist in der Lage auch mit nichtminimalphasigen Modellen zu arbeiten. Er minimiert für die Stellgrößenermittlung das Kriterium

$$J(N_1, N_2, NU, \lambda) = E[(Q_1 + Q_2)] \quad (10)$$

mit

$$Q_1 = \sum_{j=N_1}^{N_2} [\hat{y}(t+j) - r(t+j)]^2$$

$$Q_2 = \sum_{j=1}^{NU} [\lambda(j)(\Delta u(t+j-1))]^2$$

N_1 , den minimalen Vorhersagehorizont,
 N_2 , den maximalen Vorhersagehorizont,
 NU , den Stellhorizont und
 $\lambda(j)$ die Wichtung des Stellsignales darstellen.

$r(t + j)$ gibt die zukünftigen Sollwerte an.

Hierbei wird die Zeit t als diskrete Zeit betrachtet [CMT87, CB00, KLL01]. Abbildung 5 zeigt die Idee der prädiktiven Regelungen, die sich mit den folgenden Punkten skizzieren lässt.

- Zu jedem Zeitpunkt t wird mit Hilfe eines mathematischen Modells, das die Dynamik des Prozesses beschreibt, eine Vorhersage über einen langen Zeitbereich (long-range horizon) gemacht.
- Als Ergebnis dieser Vorhersagen werden verschiedene Stellgrößenverläufe generiert. Es wird aber nur die in Bezug auf den angestrebten Sollwert beste Strategie ausgewählt.
- Es wird dasjenige Stellsignal zur aktuellen Zeit t ausgegeben, welches der ausgewählten Strategie am besten entspricht. Der gesamte Vorgang wird in jedem Abtastschritt wiederholt.

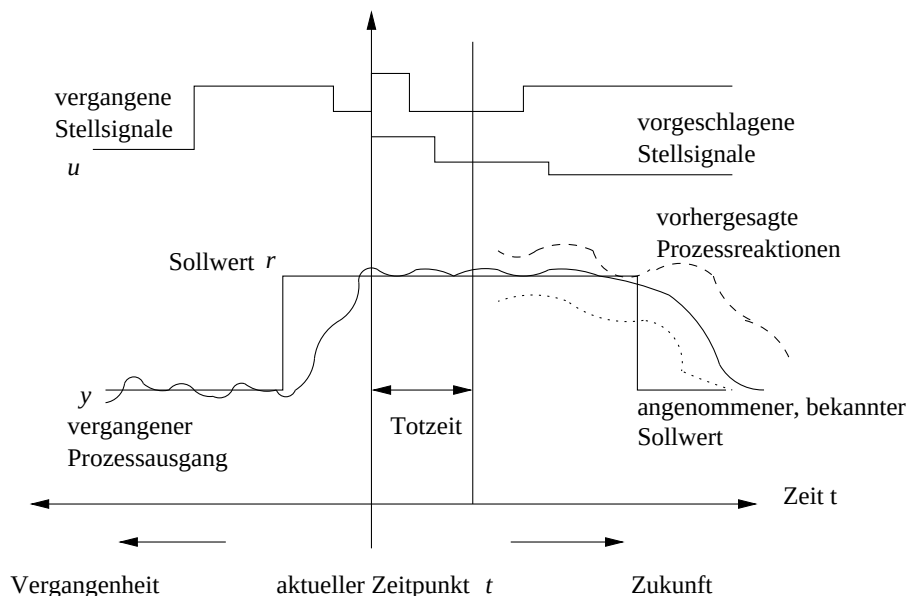


Abbildung 5: Prinzip der prädiktiven Regelung

Abbildung 6 zeigt eine intraoperative Regelung der neuromuskulären Blockade. Der Regler konnte in dem Beispiel die intraoperativen Komplikationen, die nach ungefähr 120 Operationsminuten auftraten, erfolgreich ausregeln und die neuromuskuläre Blockade in den Grenzen von $\pm 3\%$ um den Sollwert regeln.

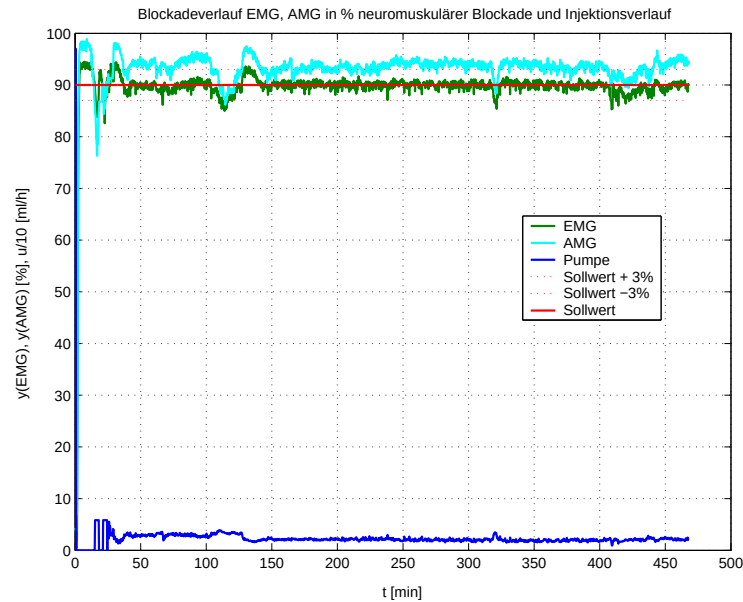


Abbildung 6: Regelung der neuromuskulären Blockade während eines chirurgischen Eingriffs

4 Zusammenfassung

Der adaptive Generalized Predictive Controller aGPC hat sich in der Simulation und klinischen Anwendung als günstig erwiesen. Er ist in der Lage, nichtminimalphasige Strecken mit variabler Totzeit und möglichen Parametervariationen zu regeln. Somit lässt er sich auch mit einfachen online-Identifikationsverfahren kombinieren.

Erfolgreiche klinische Studien an über 30 Patienten lassen für die Einsatzfähigkeit im klinischen Alltag hoffen [Sim02]. Wie Eingangs dargestellt, ist die neuromuskuläre Blockade nur ein Teil der Narkose. An der Einbeziehung der weiteren Komponenten, wie der Analgesie oder Hypnose, die dann das System zu einem Mehrgrößensystem werden lassen, wird gegenwärtig in der Arbeitsgruppe gearbeitet [Mac93, Sim02].

Literatur

- [CB00] E. F. Camacho and C. Bordos. *Model Predictive Control*. Advanced Textbooks in Control and Digital Processing. Springer, London Berlin Heidelberg New York, 2. edition, 2000.
- [CMT87] D.W. Clarke, C. Mothadi, and P.S. Tuffs. Generalised Predictive Control - Part I and II. *Automatica*, 23:137–160, 1987.
- [Hof97] R. Hofmockel. *Quantitatives neuromuskuläres Monitoring mit simultaner Anwendung der Mechano-, Elektro- und Acceleromyographie*. Habilitations-

schrift, Universität Rostock, Medizinische Fakultät, Klinik und Poliklinik für Anästhesiologie und Intensivtherapie, 1997.

- [KLL01] R. Kennel, A. Linder, and M. Linke. Generalized Predictive Control (GPC)-Ready for Use in Drive Applications ? In *IEEE Power Electronics Specialists Conference PELS*, Vancouver, Canada, 17.-22. June 2001.
- [Lin94] D.A. Linkens, editor. *Intelligent Control in Biomedicine*. Taylor & Francis, 1994.
- [Mac93] J. M. Maciejowski. *Multivariable Feedback Design*. Addison-Wesley, Cornwall, Great Britain, 1993.
- [ML98] M. Mahfouf and D.A. Linkens. *Generalised Predictive Control and Bioengineering*. Taylor & Francis, 1 Gunpowder Square, London EC4A 3DE, UK, 1998. ISBN 0-7484-0597-6.
- [Sim02] O. Simanski. *Entwicklung eines Systems zur Messung und Regelung der neuromuskulären Blockade und der Narkosetiefe*. Dissertation, Universität Rostock, Fachbereich Elektrotechnik und Informationstechnik, Institut für Automatisierungstechnik, 2002.

Das intelligente Kraftfahrzeug – mehr Sicherheit durch eine komplexe Signalverarbeitung mittels Neuro-Fuzzy-Steuerung

Prof. Dr.-Ing. Peter Roßmanek

Prof. Dr.-Ing. Christine Wahmkow

**Fachbereich Maschinenbau
Fachhochschule Stralsund**

Da Fahrzeugführer im heutigen Straßenverkehr täglich Extremsituationen ausgesetzt sind, ist die Erhöhung der Fahrsicherheit eine der bedeutendsten Aufgaben moderner Kraftfahrzeugtechnik. Das kann durch ein komplexes System der Signalverarbeitung erfolgen. Innerhalb eines Forschungsschwerpunktes am Fachbereich Maschinenbau der FH Stralsund werden folgende Teilschritte bearbeitet:

1. Intelligente Datenauswertung mittels Neuro-Fuzzy

Die Beschreibung des dynamischen Zustandes eines Fahrzeuges während der Bewegung ist hochgradig nichtlinear. Eine mathematische Beschreibung oder eine Modellierung mittels statistischer Verfahren sind aufgrund der Komplexität nur ungenügend geeignet. Bei Anwendung der Fuzzy logic in Kombination mit Neuronalen Netzen kann durch die Fähigkeit des Lernens eine hinreichend genaue Regelbasis aus dem realen Datenmaterial gefunden werden. Diese Regelbasis dient zur schnellen Auswertung der aktuellen Fahrsituation und Einstellung der Aktoren im Millisekundenbereich.

2. Auswirkungen der Signalverarbeitung auf Aktoren im Echtzeitbetrieb

Derzeit werden Fahrwerksparameter werksseitig eingestellt. Im Ergebnis der Forschung sollen die Fahrwerksparameter in Abhängigkeit vom Fahrverhalten des Fahrzeugführers automatisch durch Mikroprozessoren gesteuert werden. Das sind zur Fahrzeugstabilisierung notwendige Dämpfer in den Radaufhängungen und für die Regulierung der Motorleistung notwendige Einspritzzeiten und -mengen.

3. Fahrwerksdaten als zusätzliche Eingangsparameter zur Signalverarbeitung

Neben den allgemein üblichen Parametern wie Raddrehzahlen, Beschleunigung in Längs- und Querrichtungen und Witterungsverhältnisse (Außentemperatur und Regensensor) werden weitere Fahrwerksparameter zur Beschreibung der aktuellen Fahrzeugsituation mit zur Auswertung heran gezogen. Das sind vor allem situationsbezogene Parameter wie Gierraten, Federwege und Beschleunigungen, die am Fahrzeug selbst auftreten.

Simulation leitungsgebundener elektromagnetischer Phänomene im KFZ-Bordnetz

H. Voelker^{*}, A. Schluß^{*}

Zusammenfassung

Ausgehend von experimentellen Ergebnissen wird eine Vorgehensweise für simulative Untersuchungen zur leitungsgebundenen elektromagnetischen Verträglichkeit im KFZ-Bordnetz beschrieben. Der untersuchte Frequenzbereich reicht dabei bis 30MHz. Eine Berücksichtigung ausführlicher Ersatzschaltbilder von passiven Entstörbauelemente in der Simulation ist notwendig.

1 Einleitung

Der Einzug immer neuer Komfortfunktionen in heutige Kraftfahrzeuge wird zu einem überwiegenden Teil durch leistungselektronische Stellglieder, z.B. in sogenannten X-by-Wire-Konzepten, realisiert. Hierbei wird das statische Ein- oder Ausschalten elektrischer Verbraucher durch PWM-Steuerverfahren abgelöst. Diese leistungselektronischen Schalter generieren elektromagnetische Störungen, welche sich im gesamten KFZ-Bordnetz ausbreiten.

Die beiden an dieser Stelle vorgestellten Applikationen besitzen ein vom Arbeitspunkt abhängiges Störspannungsspektrum, das zum gegenwärtigen Zeitpunkt ausschließlich durch passive Entstörkonzepte beeinflusst wird. Entstörkonzepte auf Halbleiterbasis sind durchaus denkbar und bieten hier vielversprechende Ansatzpunkte zur Reduzierung des passiven Aufwandes.

2 Klassifizierung von Störschaltungen

Leistungselektronische Applikationen entnehmen dem Bordnetz im allgemeinen blockförmige Ströme, die Spannungsschwankungen im gesamten Bordnetz zur Folge haben. Unter EMV-Gesichtspunkten lassen sich diese Störungen, wenn sie in vergleichbaren Leistungsbereichen arbeiten, in energiearm und energiereich unterteilen. Da die Ursache der elektromagnetischen Störung der geschaltete Strom im Bordnetz ist, kann für jede der Störschaltungen, ausgehend vom realen Stromverlauf, der Strom durch Geradenapproximation nachgebildet werden.

^{*}Universität Rostock, Fachbereich Elektrotechnik und Informationstechnik
Institut für Elektrische Energietechnik, Albert-Einstein-Str. 2, 18059 Rostock

2.1 Energiearme Störschaltung

Treten in einer Störquelle nur Schaltvorgänge im Bereich von einigen hundert Hertz und langsamen Anstiegsgeschwindigkeiten (im μs -Bereich) auf, so kann im betrachteten Frequenzbereich von energiearmen Störquellen gesprochen werden. Die untersuchte energiearme Störschaltung besteht aus einem Smart Highside High Current Power Switch, welcher neben den Schutzfunktionen (ESD, Temperatur, Spannung und Strom) eine Ladungspumpe besitzt, die in die Ansteuerung des implementierten Leistungsmosfets integriert ist.

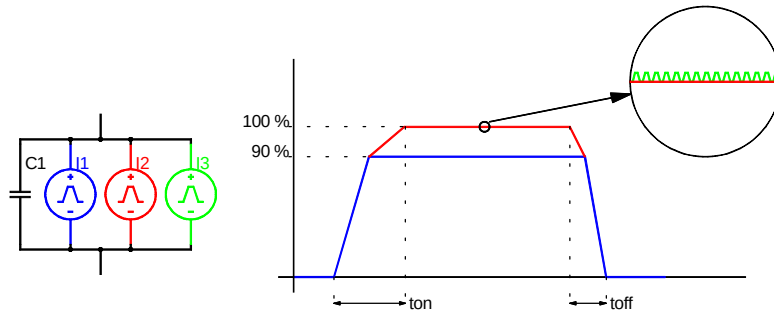


Abbildung 1: Prinzipieller Stromverlauf der energiearmen Störschaltung

Durch die Verwendung einzelner Stromquellen ist es möglich, den Einfluß der einzelnen Stromanteile im Schaltvorgang zu verdeutlichen. Eine gezielte Beeinflussung des Schaltvorgangs und seine Auswirkungen lassen sich auf diese Weise komfortabel darstellen. In Abbildung 2 sind die Spektren der einzelnen Stromanteile des Schaltvorgangs (a) und das Gesamtspektrum (b) der energiearmen Störschaltung (Leistungsschalter mit integrierter Ladungspumpe, der eine ohmsche Last mit 200Hz schaltet) dargestellt.

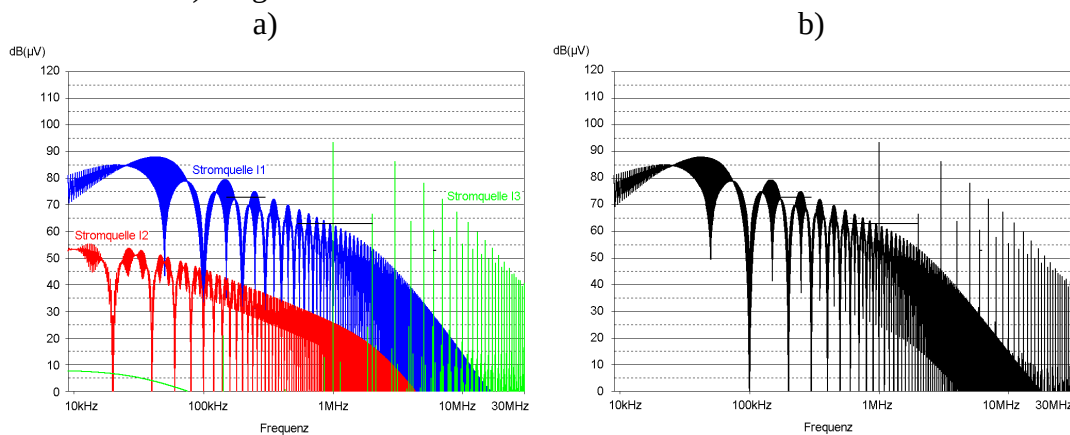


Abbildung 2: Spektralanteile der energiearmen Störschaltung

Aus Abbildung 2a ist zu entnehmen, daß das Spektrum im Frequenzbereich unterhalb der Schaltfrequenz der Ladungspumpe (1MHz) vom steilen Anstieg des Laststroms auf 90%, gegenüber dem deutlich flacheren Stromanstieg von 90% auf 100%, bestimmt wird. Weiterhin sind die Schaltfrequenz der Ladungspumpe und

deren Vielfache signifikant auszumachen, während die Stromquelle I2 keinen gravierenden Einfluß auf das Gesamtspektrum aufweist.

2.2 Energiereiche Störschaltung

Von energiereichen Störern wird bei Schaltfrequenzen von einigen Kilohertz sowie schnellen Anstiegsgeschwindigkeiten (im ns-Bereich) gesprochen, wenn diese im Vergleich zu energiearmen Störern in vergleichbaren Leistungsbereichen arbeiten. Bei der untersuchten Störschaltung handelt es sich um einen DC/DC-Steller, welcher mit 20kHz eine ohmsch/induktive Last schaltet. Diese Störer werden durch die Einführung des 42V-Bordnetzes im KFZ und den damit verbundenen anfänglichen Mischnetztopologien (14V/42V) verstärkt zum Einsatz kommen. Im Gegensatz zu der energiearmen Störschaltung werden diskrete Leistungsmosfets als Schalter eingesetzt. Die folgende Abbildung gibt diesen prinzipiellen Stromverlauf wieder, der durch das nebenstehende Simulationsmodell nachgebildet wird.

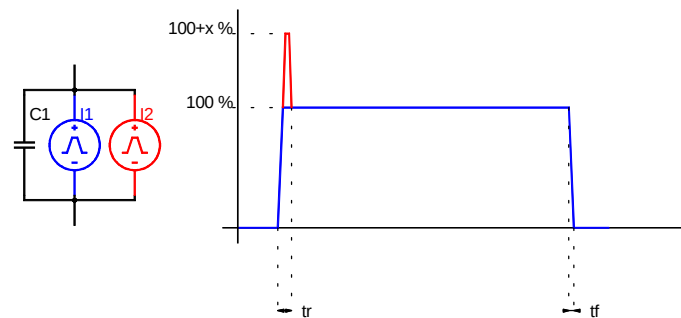


Abbildung 3: Prinzipieller Stromverlauf der energiereichen Störschaltung

Das Spektrum wird von dem durch die Stromquelle I1 nachgebildeten Stromanteil dominiert. Demgegenüber übt die im Einschaltmoment durch den Rückstromabriß der Freilaufdiode vorhandene Stromspitze keinen gravierenden Einfluß auf das Spektrum aus. Abbildung 4 verdeutlicht diese Zusammenhänge.

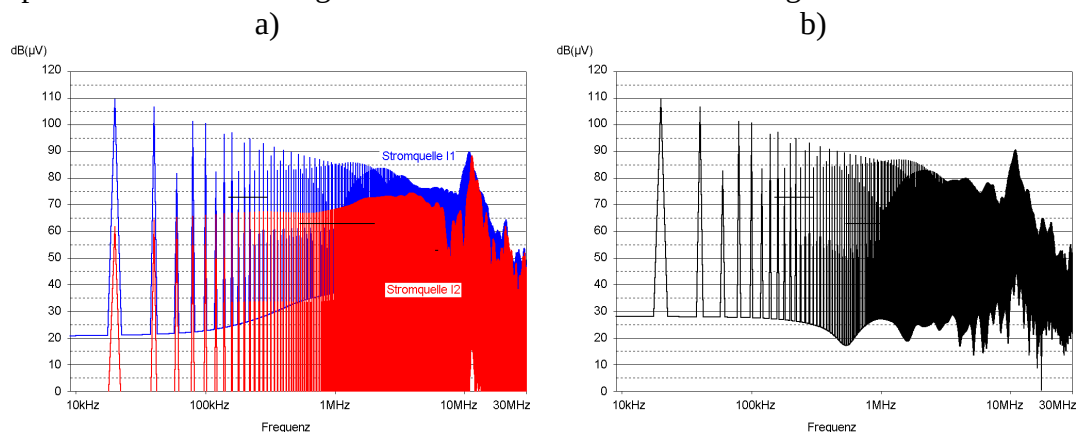


Abbildung 4: Spektralanteile der energiereichen Störschaltung

3 Simulationsmodelle passiver Bauelemente

3.1 Kondensatoren

Kondensatoren werden für die Entstörung am häufigsten eingesetzt. Ein vollständiges Ersatzschaltbild (ESB) eines Kondensators zeigt die folgende Abbildung.

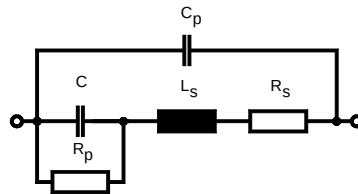


Abbildung 5: ESB eines Kondensators

Die Anschlußdrähte des Kondensators besitzen neben der parasitären Eigeninduktivität L_s zusätzlich einen ohmschen Widerstand R_s . Bedingt durch die parasitäre Eigeninduktivität des Kondensators läßt die Entstörfunktion im Hochfrequenzbereich nach. Der Kondensator zeigt mit zunehmender Frequenz ein induktives Verhalten. Auftretende Verluste im Dielektrikum werden durch den Widerstand R_p beschrieben, welcher sich im Bereich mehrerer hundert Kiloohm befindet. Eine Besonderheit ist die bei SMD-Kondensatoren zusätzlich zur Nennkapazität vorhandene Kapazität C_p , welche durch die Löt pads gebildet wird.

3.2 Drosseln

Drosseln können sowohl mit als auch ohne magnetisch leitendem Kern ausgeführt werden. Die folgende Abbildung zeigt ein Ersatzschaltbild für Drosseln mit magnetisch leitendem Kernmaterial. Durch Vernachlässigung des Widerstandes R_{Fe} ist das Ersatzschaltbild auf Drosseln ohne magnetisch leitendem Kern anwendbar.

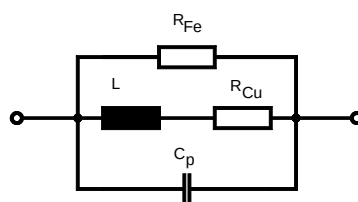


Abbildung 6: ESB einer Drossel

Die Magnetisierungsverluste bildet in dem Ersatzschaltbild der Widerstand R_{Fe} und die Kupferverluste der Widerstand R_{Cu} nach. Von besonderer Bedeutung für die Entstörfunktion ist die parasitäre Kapazität C_p , welche sich in Form von Einzelkapazitäten zwischen den einzelnen Lagen der Wicklung befindet. Diese Kapazitäten bewirken im Hochfrequenzbereich ein kapazitives Verhalten der Drossel, wodurch die Entstörfunktion nachläßt.

3.3 Stromkompensierte Drosseln

Bei stromkompensierten Drosseln wird durch die Rückführung des Laststroms über eine zweite Spule mit entgegengesetztem Wicklungssinn erreicht, daß der DC-Anteil des Laststroms nicht zur Magnetisierung des Kerns beiträgt. Die Drossel ist nur für den hochfrequenten Störstrom wirksam. Die folgende Abbildung zeigt ein Ersatzschaltbild für eine stromkompensierte Drossel.

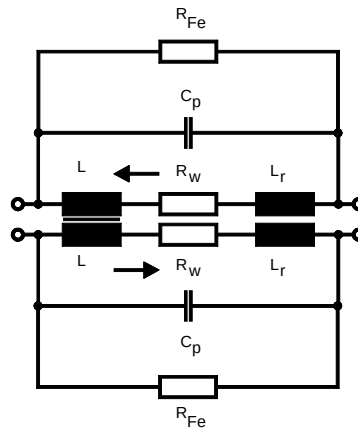


Abbildung 7: ESB einer stromkompensierten Drossel

Neben der Hauptinduktivität L ist eine Restinduktivität L_r vorhanden. Analog zu den zuvor vorgestellten Drosseln sind im Ersatzschaltbild der Kupfer- bzw. Wicklungswiderstand R_w , die Magnetisierungsverluste R_{Fe} und eine parasitäre Kapazität C_p implementiert.

4 Simulation von Störschaltungen

Die simulativen Untersuchungen zur passiven Entstörung der energiearmen Schaltung (Abbildung 8) ergaben, daß für Einhaltung der in der Norm geforderten Emissionsgrenzwerte Kondensatoren genügen. Die passive Entstörung der energiereichen Störschaltung (Abbildung 9) ist im Gegensatz hierzu aufwendiger, da neben dem Kondensator eine zusätzliche Induktivität notwendig ist. In Verbindung mit dem schaltungstechnisch bedingten Kondensator der Applikation entsteht ein sogenanntes PI-Filter. Die folgenden Abbildungen zeigen die vollständigen Simulationsmodelle für die passive Entstörung beider Störschaltungen.

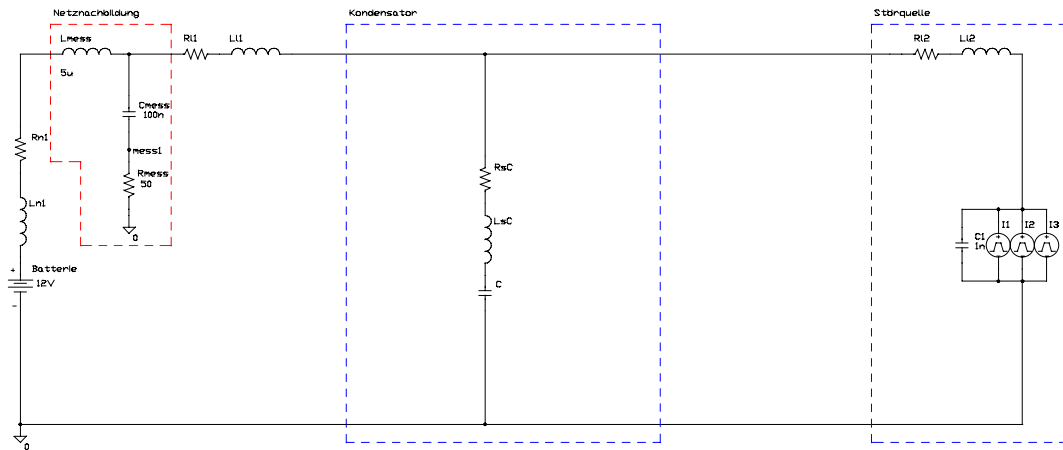


Abbildung 8: Modell zur passiven Entstörung der energiearmen Schaltung

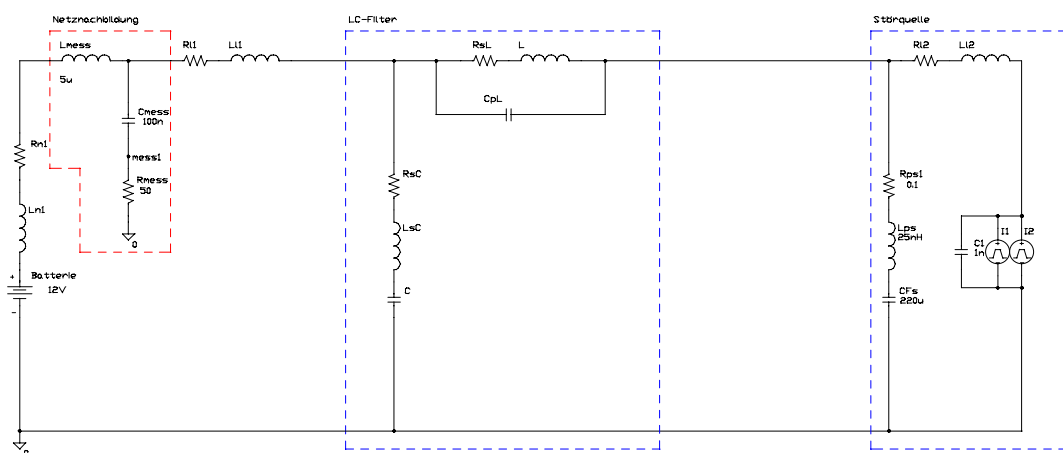


Abbildung 9: Modell zur passiven Entstörung der energiereichen Schaltung

Ausgehend von den beiden zuvor dargestellten Modellen lassen sich für unterschiedliche Arbeitspunkte die notwendigen Entstörmaßnahmen bestimmen. Im folgenden ist für beide Störschaltungen das Spektrum eines ausgewählten Arbeitspunktes vor (Ausgang) und nach der Entstörmaßnahme (passiv) dargestellt.

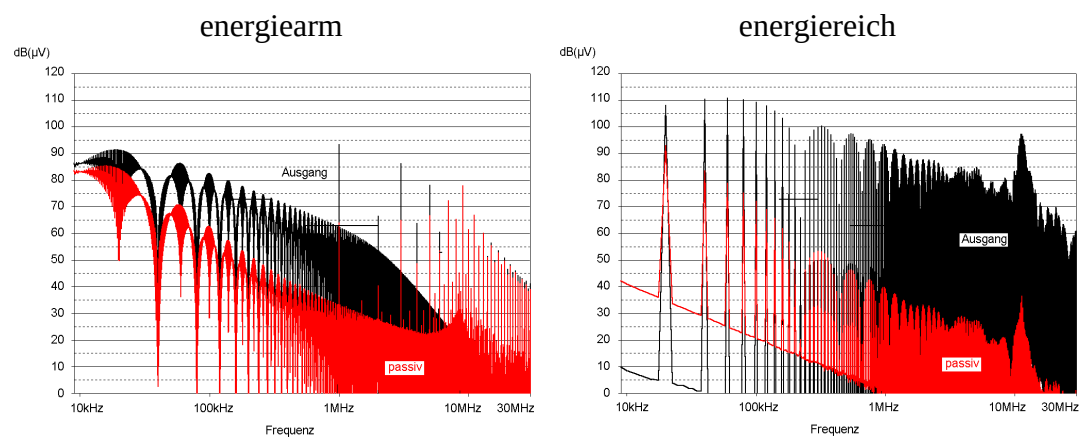


Abbildung 10: Spektren der Störschaltungen bei einem Arbeitspunkt von 25A

Für die Simulation von Entstörkonzepten direkt am bzw. im Halbleiter ist die komplette Nachbildung des Schalters mit parasitären Elementen notwendig. Am Beispiel der energiereichen Störschaltung soll dies verdeutlicht werden. Die folgende Abbildung zeigt den diesbezüglichen DC/DC-Steller.

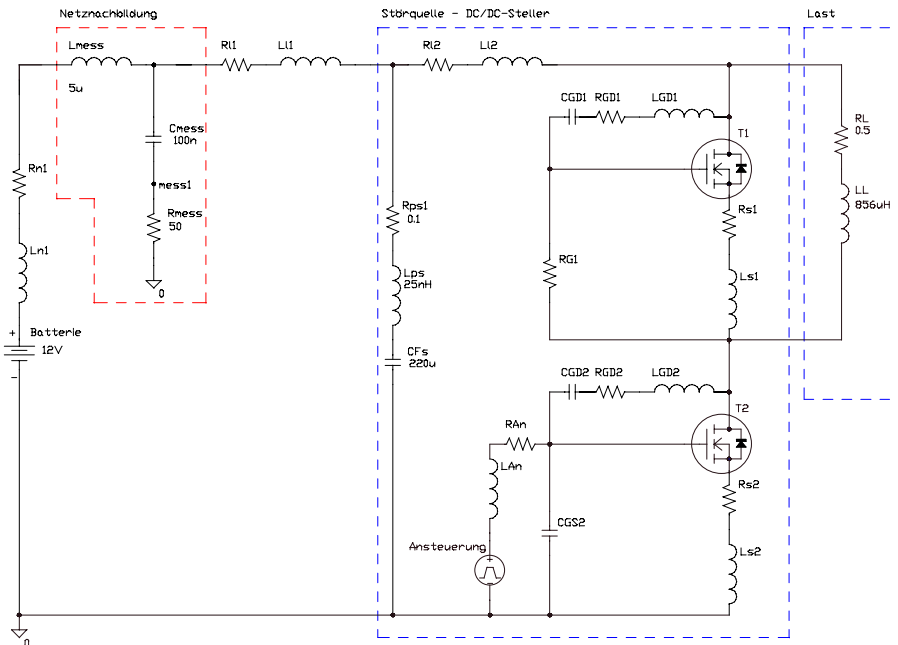


Abbildung 11: Modell zur Entstörung direkt am Halbleiter

Für die Beeinflussung des Schaltvorganges in der zuvor dargestellten Schaltung existieren im wesentlichen vier Möglichkeiten. Neben der Variation der Gate-Drain-Bauelemente (CGD2, RGD2, LGD2) bietet sich der Kondensator CGS2 am bzw. im Leistungshalbleiter für eine direkte Beeinflussung des Schaltvorganges an. Die folgenden Bilder zeigen den Einfluß der Gate-Drain-Kapazität CGD2 auf das Störspannungsspektrum der Applikation.

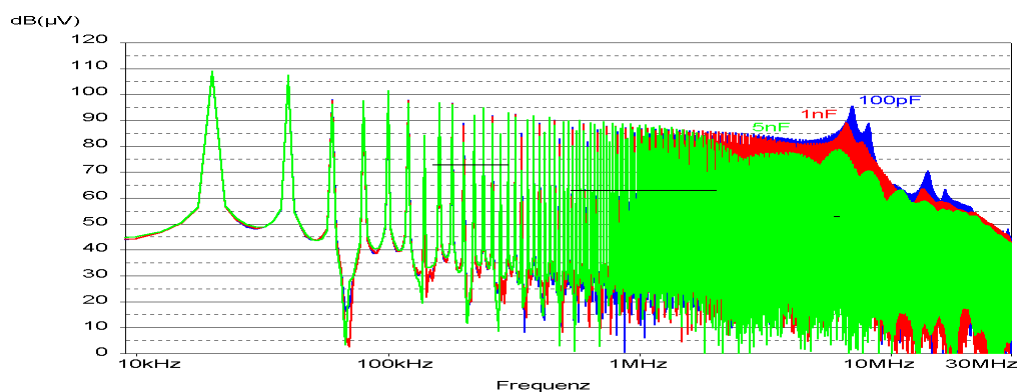


Abbildung 12: Einfluß der Gate-Drain-Kapazität auf das Spektrum

Im Gegensatz zum Gate-Drain-Kondensator CGD2 sind ein Einfluß der Gate-Drain-Induktivität LGD2 und des Gate-Drain-Widerstandes RGD2 signifikant nicht

vorhanden. Wird für die Beeinflussung des Schaltvorgangs der Gate-Source-Kondensator CGS2 herangezogen, treffen für das Spektrum die gleichen Aussagen wie für den Gate-Drain-Kondensator CGD2 zu, wenn als Ergebnis der Schaltungsmaßnahme identische Zeitverläufe in den Schaltzeitpunkten entstehen.

5 Zusammenfassung

Die beiden betrachteten Applikationen erfordern für eine normgerechte Entstörung unterschiedliche passive Schaltungskonzepte. Während sich die energiearme Störschaltung mit verhältnismäßig einfachen Mitteln in Form von Kondensatoren entstoren läßt, müssen für die energiereiche Störschaltung aufwendigere Entstörmaßnahmen mit LC-Filtern verwendet werden.

Eine andere Möglichkeit der Beeinflussung des Störspannungsspektrums wird durch die Beeinflussung des Schaltvorgangs eines Leistungshalbleiters erreicht. In diesem Fall wird durch zusätzliche Bauelemente zwischen dem Steueranschluß und den beiden Leistungsanschlüssen des Halbleiters der Schaltvorgang verzögert. Es reduziert sich das Störspannungsspektrum bei gleichzeitiger Erhöhung der Verlustleistung im Halbleiterschalter. Aufgrund der geringen elektromagnetischen Entstörwirkung dieser Variante ist sie aus dem zuvor genannten Grund nur sehr eingeschränkt zu empfehlen.

Simulation des motorischen Innenprozesses

J. Nocke, Universität Rostock

Zusammenfassung

Die Beschreibung des motorischen Verbrennungsprozesses kann zum einen mittels systemanalytischer Ansätze erfolgen, zum anderen ist eine genauere Abbildung der Prozesse mittels physikalischer bzw. empirischer Gleichungsansätze möglich. Das zur Verfügung stehende Gleichungssystem besteht aus Bilanz-, Transport- und Zustandsgleichungen sowie Ansätzen zur Beschreibung der chemischen Reaktionskinetik und der turbulenten Strömung. Mittels nulldimensionaler thermodynamischer Modelle, phänomenologischer Mehrzonenmodelle und dreidimensionaler Modelle erfolgt heute die Simulation des innermotorischen Prozesses. Die unterschiedlichen Modelle erlauben eine mehr oder weniger starke örtliche Auflösung der Vorgänge im Verbrennungsraum. Die Modelle ermöglichen bei Vorgabe des Verbrennungsdruckverlaufes die Bestimmung typischer Zünd- und Verbrennungsparameter wie Maximaldruck, Druckanstieg, Klopfverhalten, indizierter Mitteldruck, Verbrennungsschwerpunkt, u.a., sowie die Berechnung von Temperatur- und Brennstoffumsatzgeschwindigkeitsverläufen. Die Modellierung des Verbrennungsprozesses Off- oder Online bietet die Möglichkeit der Optimierung des Motorbetriebes bzgl. des Wirkungsgrades, der Abgasemission und der thermisch-mechanischen Bauteilbelastung.

Verkürzung von Entwurfszeiten durch Matlab-basiertes HPC

Rene Fink, Sven Pawletta
Hochschule Wismar

Zusammenfassung

Das Berechnungs- und Visualisierungssystem Matlab hat seit seiner Einführung in den frühen 80er Jahren eine außerordentlich starke Verbreitung in vielen ingenieur-technischen Anwendungsgebieten erfahren. Im Bereich der Regelungstechnik ist Matlab seit mehr als 10 Jahren das dominierende Standardsystem für Entwurfsprobleme.

Vor dem Hintergrund einer schnell wachsenden Komplexität vieler Entwurfsaufgaben entstand in der ersten Hälfte der 90er Jahre ein zunehmender Bedarf nach sehr hohen Rechenleistungen, die auch bei Verwendung der jeweils modernsten Prozessortechnologien durch konventionelle Rechnersysteme (*Single Instruction Single Data, SISD*) nicht mehr erbracht werden konnten. In dieser Situation erschien die Übernahme der im Bereich des Hochleistungsrechnens (*High-Performance Computing, HPC*) inzwischen weit entwickelten Technik der Parallelverarbeitung auf MIMD-Systemen (*Multiple Instructions Multiple Data*) in den ingenieur-technischen Anwendungsbereich naheliegend. Als kostengünstige Hardware-Plattform standen inzwischen PC- und Workstation-Cluster einem größeren Anwenderkreis zur Verfügung.

Wesentlich problematischer stellte sich allerdings die software-technische Seite dar. Während die Applikationsentwicklung im Bereich des Hochleistungsrechnens nach wie vor fast ausschließlich auf compilerbasierten Hochsprachen (Fortran und C) beruhte, war gerade diese Technologie im ingenieur-technischen Entwurfsbereich durch interpreterbasierte Entwicklungsumgebungen wie Matlab weitgehend abgelöst. Konsequenter Weise wurde versucht, diese Umgebungen auf Parallelverarbeitungssysteme zu portieren. So experimentierte MathWorks Inc. mit parallelen Matlab-Versionen auf MIMD-Plattformen mit gemeinsamen und verteiltem Speicher. Die erzielten Ergebnisse waren jedoch unbefriedigend, da die Matlab-internen numerischen Routinen eine zu geringe Granularität für eine effektive Parallelisierung aufwiesen [1].

Ab ca. 1994 wurde an der Universität Rostock an alternativen Methoden zur Parallelverarbeitung auf Basis Matlab-ähnlicher Systeme (*Scientific Computing Environments, SCEs*) gearbeitet. Als Ergebnis entstand der Multi-SCE-Ansatz ([2]), der einerseits erstmals eine praktikable Kombination vor paralleler und interpreter-basierter Verarbeitung ermöglichte und zum anderen auf die Ausnutzung von Parallelität auf der Anwendungsebene ausgelegt war. Da auf dieser Ebene häufig grob-granuläre Strukturen anzutreffen sind, ist eine

Parallelisierung auch auf MIMD-Systemen ohne gemeinsamen Speicher mit hoher Effektivität möglich.

Im Workshop-Vortrag wird über den aktuellen Entwicklungsstand einer Multi-SCE-Implementierung für das System Matlab (DP-Toolbox) berichtet und am Beispiel einer Parameteroptimierung für das Problem der Übergangskompensation bei Verbrennungsmotoren das Potential dieser Technik hinsichtlich der Verkürzung von Entwurfszeiten aufgezeigt ([3]).

Literatur

- [1] C. Moler. Why there isn't a parallel Matlab. *Matlab News and Notes*, page 12, Spring 1995.
- [2] S. Pawletta. Erweiterung eines wissenschaftlich-technischen Berechnungs- und Visualisierungssystems zu einer Entwicklungsumgebung für parallele Applikationen. *Fortschrittsberichte Simulation*, Band 7, ARGESIM / ASIM-Verlag, Wien, 2000.
- [3] R. Fink. Verteiltes und paralleles Rechnen im Engineering-Bereich auf Basis von Matlab. Technischer Bericht, Hochschule Wismar, 2003.

Modellbildung und Simulation für die Entwicklung von Motorsteuerungen

Autor: Krzysztof Komar (Universität Rostock)

Bei der Entwicklung neuer Konzepte für Verbrennungsmotoren werden in breitem Umfang Methoden der Modellbildung und Simulation eingesetzt. Für die verschiedenen Anwendungsbereiche existieren unterschiedliche Modellierungs- und Simulationsansätze. So werden beispielsweise in der Vorentwicklung/Verfahrensentwicklung präzise Berechnungen der Materie- und Energieströme während der Verbrennung im Zylinder benötigt. CFD-basierte Ansätze wie Fire (AVL), Star-CD (CD adapco) und KIVA (Los Alamos NL) können diese Forderung bedienen. Allerdings ist der Parametrierungsaufwand sehr hoch und das zugrunde gelegte Modell ist nicht einfach zu interpretieren oder nicht verfügbar. Auf der anderen Seite der Komplexitätsskala der Simulationsmodelle stehen HIL-Systeme wie CARTS, LabCar oder EnDyna mit einfachen Näherungen. Ein Hauptanwendungsbereich dieser Systeme ist die Funktionstestung von Steuergeräten bezüglich der integrierten Diagnoseverfahren.

Eine geringe Verbreitung haben dagegen Simulationssysteme, die speziell auf die Funktionsentwicklung und Applikation von Motorsteuergeräten orientieren. In Ermangelung solcher Systeme ist es hier üblich, für die Entwicklung einer Einzelfunktion nur ein zugeschnittenes dynamisches Teilmodell des Motors zu verwenden. Hierbei werden häufig Kopplungen mit anderen Teilsystemen vernachlässigt, die in dem zu steuernden Prozess oder durch die Steuergerätefunktionalität selbst entstehen.

Im Beitrag werden Wünsche und Anforderungen diskutiert, die aus Sicht der Funktionsentwicklung an ein solches Simulationssystem zu stellen sind. Weiterhin werden Ansätze für eine Realisierung betrachtet.

