

TWO DIFFERENT APPROACHES FOR MODELING BIOCHEMICAL REACTION NETWORKS WITH PROMOT

S. Mirschel, K. Steinmetz, A. Kremling

Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg, Germany

Corresponding author: S. Mirschel, Max Planck Institute for Dynamics of Complex Technical Systems, Systems Biology Group

39106 Magdeburg, Sandtorstraße 1, Germany, mirschel@mpi-magdeburg.mpg.de

Abstract. The modeling tool PROMOT provides an unique framework for setup, editing and visualization of systems which are relevant in chemical engineering and systems biology. PROMOT is based on a modular modeling concept which is well suited to describe modular structures of technical and biological systems. These systems can be modeled using either a quantitative or qualitative modeling approach. The quantitative approach is based on differential algebraic equations whereas the qualitative approach is a description of the system in the form of logical equations. Predefined libraries containing modeling entities for the different modeling approaches and for various application areas are provided to facilitate the model setup. Furthermore, PROMOT offers functionalities and routines for validating, exploring and visualizing biochemical reaction networks.

1 Introduction

In recent years modeling of reaction systems tend to be more complex. Especially in systems biology researchers are faced with large and highly interconnected networks that have to be mapped on various types of mathematical models. Handling of such systems requires efficient and comprehensive computing tools as well as a powerful modeling concept to describe the system under investigation.

The modeling tool PROMOT provides a framework for setup, editing and visualization of such systems which are relevant in chemical engineering and systems biology. Several excellent tools have been developed for the setup and editing of biochemical reaction networks such as CellDesigner [3], JDesigner [18] or Edinburgh Pathway Editor [20] widely used in the systems biology community or gPROMS ModelBuilder [1] dedicated to chemical engineering. However, in contrast to these tools PROMOT provides two features – modular modeling and support of two different modeling approaches using the same graphical user interface.

The concept of modular modeling with PROMOT is successfully applied to the study of diverse systems in chemical engineering including reaction and separation systems, membrane reactors [13], catalytic reactors and fuel cells [6]. Furthermore, PROMOT is applied to intracellular reaction systems [12, 11] using either differential algebraic equations (DAEs) or, more recently, a logical (Boolean) modeling formalism.

The paper is organized as follows: In Section 2 the basic concepts of PROMOT are introduced. The section is particularly focusing on the network theory, the modular modeling approach and several GUI components which facilitate the interaction of the user with the individual system to be modeled. These theoretical concepts are the principal foundation for the two different approaches for modeling biochemical systems in PROMOT. In Section 3 and Section 4 the modeling approaches are described in detail. For each modeling approach PROMOT is being extended with a special library providing basic modeling entities. The different modeling libraries are introduced and for the particular approach a model example is presented. Finally, a summary is given in Section 5.

Detailed functions and features of PROMOT are summarized in Table 1. Installation instructions, tutorials and up-to-date versions can be found at <http://www.mpi-magdeburg.mpg.de/projects/promot/>.

2 Basic Concepts

PROMOT is based on a modular modeling concept which is well suited to describe modular structures of technical and biological systems. This is beneficial in order to model systems on different levels and granularity. The elementary modeling entities can be aggregated to higher structural elements like chemical apparatus or overall pathways in systems biology. To optimally address this feature, PROMOT utilizes object-oriented and equation-based modeling. As it is shown later, the modular modeling concept was applied to quantitative models and has also been extended to include qualitative models, frequently used in systems biology.

The basic theoretical foundation of the modular modeling concept is the network theory [4] that provides ways to decompose various processes into hierarchical units in a systematic manner. The hierarchical structure of the process is represented in several levels. For setting up this hierarchy two basic types of elements are provided, namely *components*, representing the holdup of different physical quantities, and *coupling elements* describing the interactions and transports between the different components. Components provide information about their

<i>Type/Name of Criterion</i>	<i>Value of Criterion</i>
Capabilities	
Setup, Editing	Yes
Simulation	No
Analysis	Yes
Database	Yes, model library
Import	SBML, CELLNETANALYZER,
Export	SBML, CELLNETANALYZER, DIVA, Diana, MATLAB
Modeling Approaches	
Ordinary differential equations (ODE)	Yes
Differential algebraic equations (DAE)	Yes
Partial differential equations (PDE)	No
Stochastic	No
Events	Yes
Logical	Yes
Representations	
User interfaces for model definition	Diagrammatic, Textual
Graphical representation of multiple compartments	Yes
Representation of interface reactions	Yes
Representation of multiple reactant types	Yes
Representation of regulatory interactions	Yes
Automated network layout	Yes
Library of predefined rate expressions	Yes
Hierarchy	Yes
Requirements	
Website	Yes, http://www.mpi-magdeburg.mpg.de/projects/promot/
Hardware requirements	~ 200 MB disk space
Software requirements	Java (JRE 6), Common Lisp
Platforms	Windows, Linux
Availability	
Open source	Yes, LGPL license
Free for academic use	Yes
Free for commercial use	Yes

Table 1: Summary of requirements and features of ProMoT according to criteria used in [17] and [19], addressing system requirements, supported modeling frameworks and representations or availability of the modeling tool. PROMOT supports SBML Level 2 Version 1, except events.

potentials, that is their concentration and require information about the fluxes coming in and leading out of them. Essentially they balance the potentials with regard to the fluxes. Coupling elements calculate the fluxes, that is the reaction rates, depending on the potentials and provide the flux information to the components. This two-directional information exchange of potentials and fluxes forms a potential-flux vector. In systems biology, the top level is represented by a bio-reactor system with feeds and drain. The reactor again consist of phases (e. g. liquid and bio-phase) that are coupled by phase-boundaries or membranes and finally the phases consist of storages that are coupled via reactions or diffusive and convective relations. The liquid-phase model comprises storages for the substrates (concentrations c_l (unit $[g/l]$)) and uses the volume as an extensive reference quantity V_l (unit $[l]$). The bio-phase contains storages for intracellular metabolites, and the biochemical reactions as coupling elements. As an extensive reference quantity, here, the biomass m_{bio} is used.

In PROMOT setting up and managing models can be done using either a textual or diagrammatic user interface. The latter is more convenient because a systems can be described by many heterogeneous, interconnected and nested components and therefore a graphical setup can give the modeler an improved perception of the model and its inherent structure. This results in a better understanding of the model, furthermore in the reduction of modeling errors and time saving in the modeling process.

The model setup and editing can be done in a convenient and user-friendly manner in the PROMOT Visual Editor. The different modeling entities can be placed on a construction area using drag&drop from the PROMOT Visual Browser. Afterwards the elements can be connected by links to achieve an aggregation hierarchy. Modeling entities can be obtained either from predefined modeling libraries, by deriving from already existing entities or by creating from scratch. The behavior of the modeling entities (defined by variables) can be altered by the PROMOT Variable Editor. For the exploration and the visualization of a modular model the PROMOT Visual Explorer can be launched. Here, context sensitive zooming techniques are used to navigate the individual modules in the context of the whole model. In this way, the user gets an overview of the model. Furthermore, visualization methods are provided which are able to graphically abstract and modify the graphical representation of the model. This enables the user to generate customized representations of the biochemical reaction network.

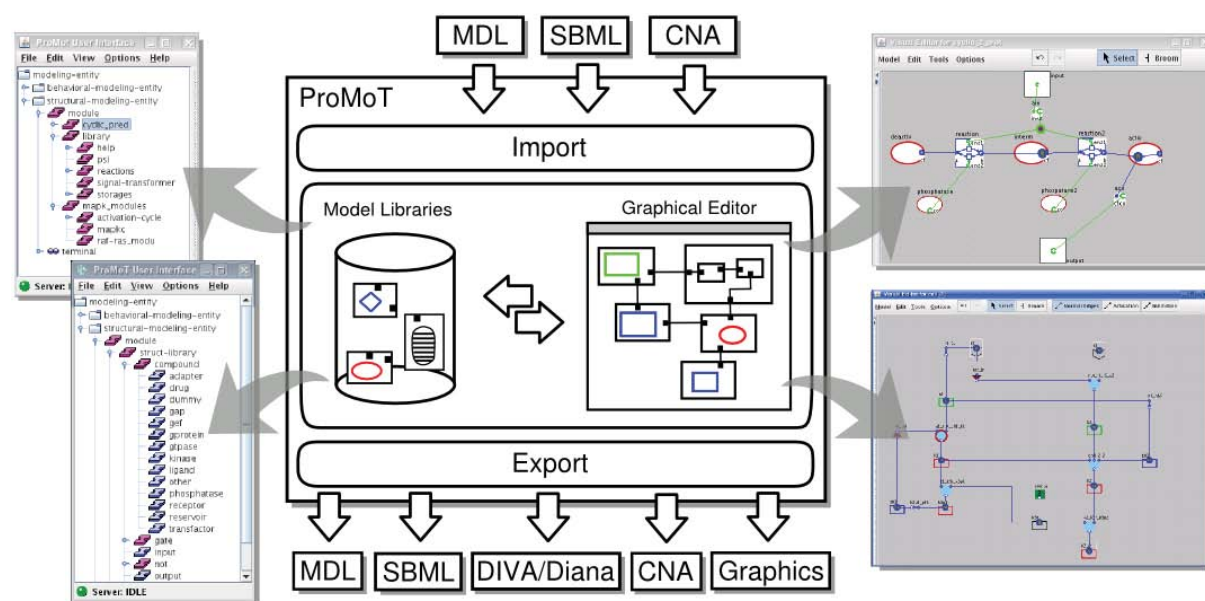


Figure 1: Modeling with PROMOT is facilitated by modeling libraries (left) and diagrammatic user interfaces (right). Several options for the import and export of PROMOT models are available (MDL – Modeling Definition Language, SBML – Systems Biology Markup Language, DIVA/Diana – DIVA or Diana models, CNA – CellNetAnalyzer format, Graphics – various bitmap and vector formats).

In order to analyze or simulate models, PROMOT is tightly coupled with several simulation and analysis tools in order to get a closed environment which results in an optimized modeling workflow. With respect to the reaction system, PROMOT can read three different formats – MDL, SBML and CNA (see Figure 1). The Model Definition Language (MDL) is a declarative, object-oriented language used by PROMOT, in particular as storage format of libraries and models. The Systems Biology Markup Language (SBML) [7] represents biochemical reaction systems. SBML is based on XML and the standard exchange format for models applied in systems biology. Thus, PROMOT is linked with many other tools via SBML. Since SBML at its present state includes no layout information of the network representation, PROMOT provides several automated layout algorithms. The CNA format is used for the storage of logical models described in CELLNETANALYZER [9] and as an exchange format of logical models.

The common approach for modeling biochemical reaction systems is kinetic modeling using mass balance equa-

tions. However in systems biology, the complexity of the biochemical networks under investigation suggests to describe the systems not only by well known mass balance equations but also in a more qualitative way by means of logical operators. In the contribution at hand, it is focused on quantitative as well as qualitative modeling approaches that are prevalently used in systems biology and that are available in PROMOT. Both modeling approaches benefit from the modular modeling concept used in PROMOT.

3 Modeling based on DAE systems

The quantitative modeling approach in PROMOT has its origin in chemical engineering and is also capable to describe cellular processes in systems biology such as intracellular metabolic or signaling processes [5]. The resulting equations that allow to simulate the system are based on mass balances and therefore are represented by a set of ordinary differential equations (ODE system). However, in many cases, model reduction can be performed that allow to reduce the number of dynamic state variables by setting up algebraic relationships between the compounds. For instance, the processes of signal transduction are in general much faster than the processes of protein synthesis and therefore its dynamics can be neglected.

Elementary modeling entities that are used in the quantitative description are substance storages, representing compounds (red elliptical symbols), substance transformers, representing biochemical reactions (blue square symbols), and signal transduction units (green rectangular symbols).

An example: the *lac* operon in *E. coli*

As an example, the very well known uptake system for lactose is shown in Figure 2. The storages considered

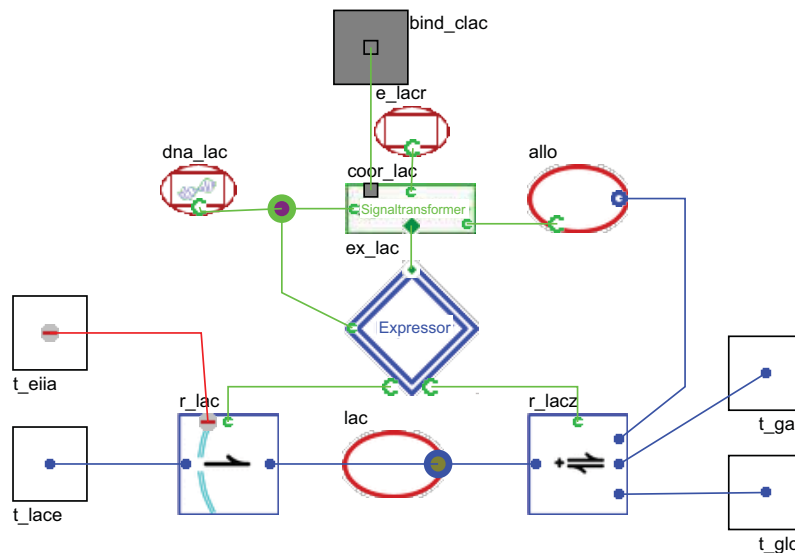


Figure 2: Setup of a model for lactose transport as it appears in the PROMOT Visual Editor. The whole drawing is the interior of a module. The white and gray boxes at the outer edges are external terminals (in this case referring to terminals of the submodules). The elements inside the drawing are aggregated submodules, edges represent links [5].

are internal lactose and allolactose that are described both with mass balance equations. The mass balances are converted to equations for the concentrations:

$$\dot{c}_{lac} = r_{lac} - r_{lacz} - \mu c_{lac} \quad (1)$$

$$\dot{c}_{allo} = r_{lacz} - \mu c_{allo} \quad (2)$$

The expression module (blue double bordered diamond symbol) describes the synthesis and degradation of the respective enzymes *LacY* and *LacZ*:

$$\dot{c}_{LacY} = r_{syn1} - r_{deg1} - \mu c_{LacY} \quad (3)$$

$$\dot{c}_{LacZ} = r_{syn2} - r_{deg2} - \mu c_{LacZ} \quad (4)$$

where the rate of synthesis depends on the inputs from the signal transduction unit. For degradation, simple mass action laws are applied. In the transformer module (green rectangular symbol) processes of signal transduction are considered. Inputs are the concentration of the inducer allolactose c_{allo} , the concentration of the repressor, e_{lacr} (taken as a constant value) and the number of the corresponding DNA templates dna_{lac} . Output is the fraction of occupied promoters with transcription factors. This fraction can be taken as a measure to describe the rate of synthesis of the proteins. Furthermore, an input $bind_{clac}$ from a different unit, the “modulon” unit, taking into

account control by a global transcription factor, here, *Crp*, is considered. All inputs are processed in the signal transduction unit. The signal transduction unit is represented by a set of two algebraic equations that sum up all the different conformations for the repressor and the DNA binding sites.

$$c_{DNA0} = c_{DNA} + \sum_K c_{KDNA}^K \quad (5)$$

$$c_{LacR0} = c_{LacR} + c_{DL} \quad (6)$$

In the example, the repressor *LacR* is either in the free form or bound with alloctose c_{DL} .

Overview on the modeling library for DA systems

The modeling library for DA systems is organized as follows: Besides some miscellaneous classes¹, the main classes in the library are the basic modeling entities for substance storages, substance transformers and signal transduction. Specified modules for *E. coli* can be found in the class “modell_ecoli” while the model for *E. coli* is stored in the class “ecoli”. The overall model, including the liquid and the bio-phase is stored in the class “reactor”. The class “modell_ecoli” contains subclasses for metabolic pathways, in particular metabolic blocks that are regulatory units with different pathways, and metabolic units representing larger parts of the metabolism like catabolic or anabolic units. Figure 3 shows screenshots from the PROMOT Browser with loaded libraries. Table 2 summarizes the classes and the properties of the current library for DA systems. Models that are set up

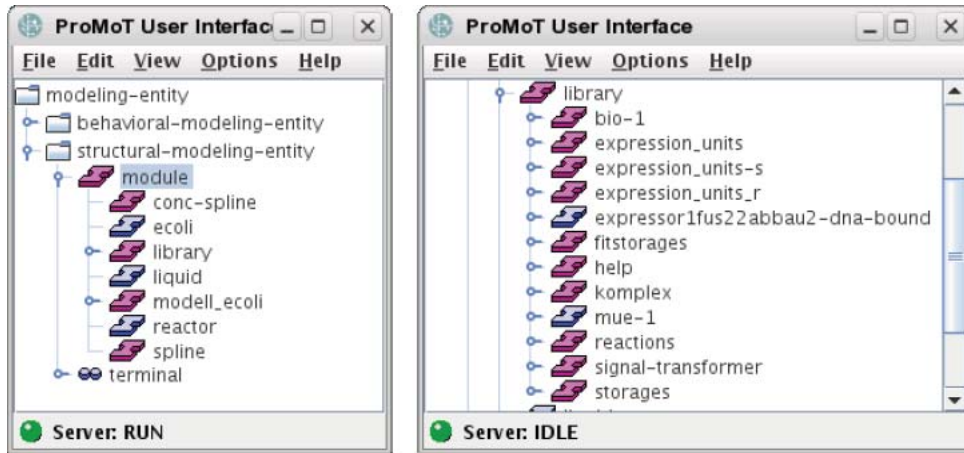


Figure 3: The PROMOT Browser showing the status quo of the modeling library for DA systems. All loaded classes including the model and defined submodules (left), classes of the library (right).

Table 2: Summary of classes loaded in the PROMOT Browser. ^aComplex reactions are e. g. group translocation processes like the phosphotransferase system.

<i>E. coli</i> model	
ecoli	model for <i>E. coli</i>
liquid	model of the liquid phase
modell_ecoli	all modules for <i>E. coli</i>
reactor	overall model
The library	
gene expression	43 variants
reactions (complex units ^a)	4 variants
reactions (enzyme catalyzed)	163 variants
reactions (mass action)	43 variants
reactions (polymerization)	3 variants
signal transduction	27 variants
storages	25 variants

with PROMOT finally can be exported to various simulation engines such as MATLAB, DIVA or its successor Diana [10]. Furthermore, a different set of kinetic parameters can be reloaded in PROMOT, e.g. after parameter estimation with DIVA.

¹The items of libraries are called ‘classes’ since models and libraries in PROMOT are built up in an object-oriented manner. Subclasses of a certain class can be easily defined. The object-oriented concept in PROMOT [5] can be useful to set up complex libraries or handle model variants.

A very detailed model for *E. coli* that was set up with the help of PROMOT is described in several publications, see [2] and references therein. It describes uptake of six carbohydrates (glucose, lactose, glycerol, galactose, sucrose, and glucose 6-phosphate). The pathways for the individual carbohydrates include also a description of protein synthesis. Gene expression is considered on different levels. The individual modules include a description of the specific control by the repressor (“regulon level”). In general, the repressors like *LacI* have a small number of binding sites on the genome. Furthermore, all pathways considered in the model are under control of the global transcription factor *Crp* (“modulon level”). To calculate the influence of the two transcription factors, a method introduced in [12] was applied. The method defines a hierarchy where the signal is transduced from the upper level, here the modulon level to the regulon level, but not vice versa.

To simulate the dynamics of cellular systems it is desirable to determine kinetic parameters from experimental data. Since in most cases a direct measurement is not possible, the parameters are estimated during a parameter identification procedure. This comprises the check of identifiability and the estimation of the parameters. With the model, kinetic parameters for a detailed dynamic model of carbohydrate uptake were estimated. Model predictions were verified by measuring time courses of several extra- and intracellular components such as glycolytic intermediates (in a pulse experiment), EIIA phosphorylation level, both LacZ and PtsG concentrations and total cAMP concentrations under various growth conditions. The entire database consists of 18 experiments performed with 9 different strains (wild type and mutant strains). The model describes expression of 17 key enzymes, 38 enzymatic reactions and the dynamic behavior of more than 50 metabolites. Based on the experiments and with help of the DIVA environment incorporating highly sophisticated methods for sensitivity analysis, parameter analysis and parameter estimation, 50 parameters (34 %) could be estimated. Especially the analysis of mutant strains offers the possibility to check if the control structures are reproduced well. In addition, pulse experiments, “disturbed” batch experiments and continuous culture allow to determine and analyze the dynamics in different time windows.

4 Modeling based on a logical modeling formalism

Besides the quantitative approach PROMOT supports a logical (Boolean) modeling formalism. This approach, elaborated in [8], gains increasing importance due to the lack of knowledge about detailed quantitative data of biological systems to be modeled with dynamic approaches. Often a high level of uncertainty regarding kinetic laws and parameters is observed. In this case a predictive dynamic model is not practicable. In contrast, the logical approach is a qualitative description of the system in the form of logical equations (see Table 3, column “Description”). Furthermore the approach is well suited to describe systems with large numbers of components and considerably less information about the dynamic behavior. Moreover, experimental data may be available only at discrete values (“a protein is present or not” or “if protein *A* is active then also protein *B* is active”). In this case, a qualitative approach is more appealing. For the setup of such models the literature at least in the area of systems biology offers plenty of information, which is also better comparable than kinetic parameters, measured under different conditions.

Name	Description	Def	Sh	t	x	y
r1	= 1 r1	0	0	1	122	78
r2	= 1 r2	0	0	1	197	78
r3	1 a1 + 1 !ph1 = 1 k1	#	1	1	182	439
r4	1 a1 = 1 ph2	#	0	2	293	345
r5	1 k3r + 1 k1 = 1 k3p1	#	0	1	193	570
r6	1 k2 + 1 k3r = 1 k3p2	#	0	1	544	661
r7	1 k3p1 = 1 ph1	#	0	2	177	606
r8	1 r2 + 1 !r1 = 1 a2	#	0	1	556	323
r9	1 r1 = 1 a1	#	0	1	253	220
r10	= 1 k3r	1	0	1	422	568
r11	1 a2 + 1 k1 = 1 k2	#	0	1	150	229
r12	1 ph2 + 1 a2 = 1 k2	#	0	1	294	229
r13	1 tf1 =	#	0	1	57	180
r14	1 k3p2 + k3p1 = 1 tf1	#	0	1	58	93

Table 3: The table shows an excerpt of the file containing the *toymodel* exported to CELLNETANALYZER [8]. Each row represents an individual reaction in the network. Columns define the name of the reactions, the description of the logical equations and different properties such as Def, Sh or t. Def means default value, where '#' encodes a non-determined value, otherwise the determined default value is given. Sh represents an option whether an element possesses an incomplete truth table (value 1) or not (value 0). The t property means time scale, and x and y the geometric positions in the diagrammatic map. In CELLNETANALYZER an AND connection is encoded via '+' and a NOT via an exclamation mark '!', respectively.

Overview on the modeling library for logical models

In recent years a library containing basic modeling entities for the logical modeling approach has been developed and constantly improved (see left hand side of Figure 1 and [14]). This library is primarily used to build up signal transduction networks. The modeling library contains two main classes, “compound” representing a state or species, and “gate” defining a logical interaction between compounds.

The class “compound” has been subclassed into different classes such as “receptor”, “kinase”, “adapter” or “reservoir”, which are all mathematically equivalent to “compound” but can be specifically considered in the later visualization process. To define different logical connections among the species the class “gate” is subclassed into the class “activ” (to describe a causal one-to-one relation between two compounds), the class “and” (to define the requirement of several elements to activate a certain compound), and the class “not” (to express a negative effect, i. e. an inhibition). An “or” gate can be implemented by including several “activ” elements pointing at a certain compound. Since any logical connection can be described as a combination of “ands”, “ors”, and “nots”, the set of basic modeling entities described above allows to set up any logical network of arbitrary size. In addition, to describe cases where the logic is unclear, the class “somehow” which represents logical gates with partially incomplete truth tables has also been included. Finally, the classes “source” and “drain” allow to define the boundary of the model, in particular incoming and outgoing signals.

To extend the formalism of logical networks, properties can be easily added to the different modeling entities [8]. For example, several parameters are defined such as `default_value` and `time_scale` which are exported with the model (see Table 3, columns “Def” and “t”). Multiple levels (i. e. discretizing the states into more than two levels, 0 and 1) is also implemented using the properties of the gates: Inputs and outputs of all gates possess a parameter `multiple_level` (with default value 1) encoding the level. The parameter value of the input defines which state must reach the start node to activate the target, and the parameter value of the output the level the target will reach. Furthermore the classes “compound” and “gate” define the property `ignore_in_export`. Setting the value of this property from 0 to 1 means the element would not be considered in the export to CELLNETANALYZER. In contrast the property `logical_exclude` with the value 1 instead of the default value 0 gives an option to export an element but excluded it from the analysis in CELLNETANALYZER. The property `degree_of_confidence` is used to set a level of how confident or well supported a certain reaction is. The value 1 being the most established. Besides the class “somehow”, the property `degree_of_confidence` represents an alternative concept to handle uncertainty. Additionally, all modeling entities have a property `documentation`, which can also be exported with the model.

An example: the toymodel

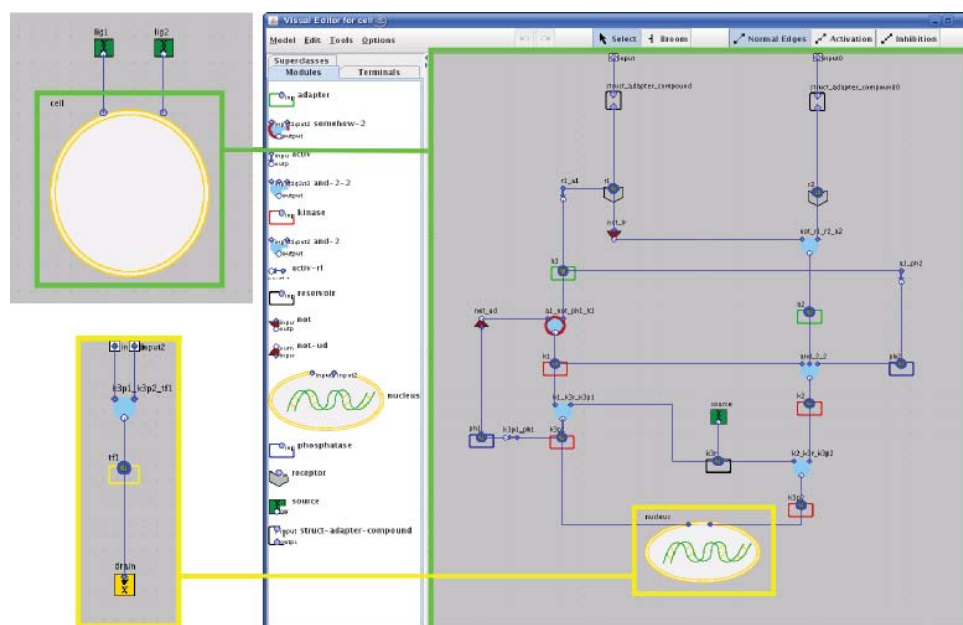


Figure 4: The logical toymodel in the PROMOT Visual Editor with the modules “cell” and “nucleus” included.

An example for such a logical network using modularity is shown in Figure 4. The top level of the model includes a module “cell”, which gets two ligands as inputs. The “cell” level contains various compounds, in particular “kinases”, “phosphatases” and a “reservoir”. Reservoirs are used to express several pools of a molecule (pools that differ by location or regulation). Furthermore the “cell” includes several gates which link different compounds by logical operations, and “not” elements realizing i. e. negative feedback loops (see Figure 4 for an example). A complex module “nucleus” which represents the nucleus of the cell is also integrated. The module “nucleus”,

again, is composed of a gate, a compound and a drain as output.

In order to analyze the logical models built up in PROMOT, the export to CELLNETANALYZER is provided in the form of logical equations (see Table 3) coupled with a graphical representation.

CELLNETANALYZER can perform different structural and functional analysis based on the logical formalism in combination with graph-theoretical methods applied to the underlying interaction graph (e. g. intervention strategies in large signaling networks or logical steady state analyses [9]). The analysis results of CELLNETANALYZER can be re-imported in PROMOT and visualized in an intuitive way. The PROMOT GUI integrates more and more functionality for efficient and user-friendly setup of logical models such as short-cuts for efficient and less error-prone work, validation for structural modeling errors while editing and special visual representations which are used to represent the logical model. This is of great value in order to interpret and discuss specific results in interdisciplinary teams.

The logical modeling approach is successfully applied to a number of projects dealing with signal transduction networks. The T-Cell Receptor Signaling [15] pathway has been modeled using 94 species and 123 interactions. More recently, the model of the Epidermal Growth Factor Receptor [16] has been developed containing 103 species, 203 interactions, 13 inputs (e. g. ligands) and 13 outputs (e. g. transcription factors).

5 Conclusions

PROMOT helps modelers in the setup, editing and visualization of biochemical reaction networks. Hereby, the modeler can choose between a quantitative or qualitative approach which are accessible by the same graphical user interface. Both modeling approaches benefit from the modular modeling concept implemented in PROMOT. For the modeling of chemical and cellular systems different predefined modeling libraries are provided to ease the procedure of model setup and editing. The scope of application of PROMOT is shown in several modeling projects dealing with chemical apparatus, metabolic pathways or signal transduction networks.

6 Acknowledgements

This work is financially supported by the German Ministry of Education and Research (BMBF) through the HepatoSys and ForSys initiative.

7 References

- [1] Asteasuain, M. et al.: *Modeling and optimization of a high-pressure ethylene polymerization reactor using gPROMS*. Computers & Chemical Engineering, 32(3) (2008), 396-408.
- [2] Bettenbrock, K. et al.: *A quantitative approach to catabolite repression in Escherichia coli*. J. Biol. Chem., 281(5) (2006), 2578-2584.
- [3] Funahashi, A. et al.: *CellDesigner 3.5: A Versatile Modeling Tool for Biochemical Networks*. Proceedings of the IEEE, 96(8) (2008), 1254-1265.
- [4] Gilles, E. D.: *Network theory for chemical processes*. Chemical Engineering Technology, 21 (1998), 121-132.
- [5] Ginkel, M. et al.: *Modular modeling of cellular systems with ProMoT/Diva*. Bioinformatics, 19(9) (2003), 1169-1176.
- [6] Hanke, R. et al.: *Application of hierarchical process modelling strategies to fuel cell systems - towards a virtual fuel cell laboratory*. Fuel Cells – From Fundamentals to Systems, 5(1) (2005), 133-147.
- [7] Hucka, M. et al.: *The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models*. Bioinformatics, 19(4) (2003), 524-531.
- [8] Klamt, S. et al.: *A methodology for the structural and functional analysis of signaling and regulatory networks*. BMC Bioinformatics, 56(7) (2006).
- [9] Klamt, S. et al.: *Structural and functional analysis of cellular networks with CellNetAnalyzer*. BMC Systems Biology, 2(1) (2007).
- [10] Krasnyk, M. et al.: *The ProMoT/Diana Simulation Environment*. Proc. 16th European Symposium on Computer Aided Process Engineering, (2006), 445-450.
- [11] Saez-Rodriguez, J. et al.: *Dissecting the puzzle of life: modularization of signal transduction networks*. Comput. Chem. Eng., 29(3) (2005), 619-629.
- [12] Kremling, A. and Gilles, E. D.: *The organization of metabolic reaction networks: II. Signal processing in hierarchical structured functional units* Metab. Eng., 3(2) (2001), 138
- [13] Mangold, M. et al.: *A model library for membrane reactors implemented in the process modelling tool ProMoT*. Comput. Chem. Eng., 28 (2004), 319.
- [14] Saez-Rodriguez, J. et al.: *Visual setup of logical models of signaling and regulatory networks with ProMoT*. BMC Bioinformatics, 7 (2006).
- [15] Saez-Rodriguez, J. et al.: *A logical model provides insights into T-cell signaling*. PloS Comp. Biol., 3(8) (2007), 163.

- [16] Samaga, R. et al.: *The logic of EGFR/ErbB signaling: Theoretical properties and analysis of high-throughput data*. In preparation.
- [17] Alves R. et al.: *Tools for kinetic modeling of biochemical networks*. Nature Biotechnology, 24 (2006), 667-672.
- [18] Sauro, H. M. et al.: *Next Generation Simulation Tools: The Systems Biology Workbench and BioSPICE Integration* OMICS: A Journal of Integrative Biology, 7(4) (2003), 355-372.
- [19] SBML.org: http://sbml.org/SBML_Software_Guide/SBML_Software_Matrix. Systems Biology Markup Language website, last visited: 20 December 2008.
- [20] Sorokin, A. et al.: *The pathway editor: a tool for managing complex biological networks*. IBM Journal of Research and Development, 50(6) (2006), 561-573.