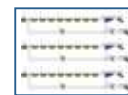
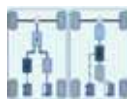
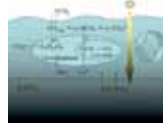
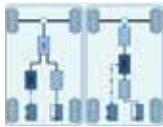
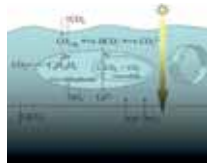
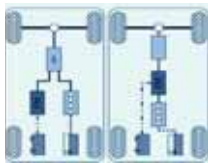
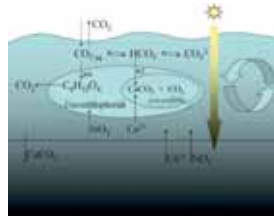
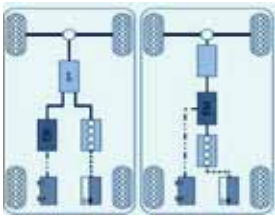


# SNE SIMULATION NOTES EUROPE



**EUROSIM 2013 - 8th EUROSIM  
Congress on Modelling and Simulation**  
Cardiff, September 10-13, 2013  
[www.eurosim2013.info](http://www.eurosim2013.info)



Volume 22 No. 3-4, Dec. 2012  
doi: 10.11128/sne.22.1014

Print ISSN 2305-9974  
Online ISSN 2306-0271



Journal on Developments and  
Trends in Modelling and Simulation  
Membership Journal for Simulation  
Societies and Groups in EUROSIM



*515.000.000 KM, 380.000 SIMULATIONEN  
UND KEIN EINZIGER TESTFLUG.*

*DAS IST MODEL-BASED DESIGN.*

*Nachdem der Endabstieg der beiden Mars Rover unter Tausenden von atmosphärischen Bedingungen simuliert wurde, entwickelte und testete das Ingenieur-Team ein ausfallsicheres Bremsraketen-System, um eine zuverlässige Landung zu garantieren. Das Resultat – zwei erfolgreiche autonome Landungen, die exakt gemäß der Simulation erfolgten. Mehr hierzu erfahren Sie unter: [www.mathworks.de/mbd](http://www.mathworks.de/mbd)*

**MATLAB<sup>®</sup>  
& SIMULINK<sup>®</sup>**

## Editorial

Dear Readers – This last issue of SNE Volume 22 finalizes the essential changes for SNE in the years 2011 and 2012 – from Simulation News Europe to Simulation Notes Europe, from limited SNE access to an open access strategy for Online SNE, from classical citation to introduction of DOI identifiers for web availability and citation index purposes. ARGESIM, SNE's official publisher, is now member of CrossRef, the official DOI® link registration agency for scholarly and professional publications. CrossRef's citation-linking network today covers tens of millions of articles from thousands of scholarly and professional publishers.

ARGESIM manages assignment of DOI identifiers for SNE contributions (all types of Notes) as well as for SNE issues, and manages the DOI registration at CrossRef. SNE contributions from SNE 21(1) on got already DOI's (ARGESIM DOI prefix, publication (SNE), SNE note type, 5-digit number, e.g. 10.11128/sne.22.tn.10137), this SNE issue SNE 22(3-4) comes first with an issue DOI (using a four-digit classification, 10.11128/sne.22.1014). It is intended to assign DOI's to contributions in SNE back issues.

This issue publishes contributions with emphasis on process engineering, bioengineering, and mechatronic engineering, partly extended and/or revised contributions from MATHMOD Vienna Conference Series and from ASIM / EUROSIM Workshops, discussing e.g. specific modelling and identification problems in these areas, and as common denominator problems of DAE

modelling and DAE simulation.

I would like to thank all authors for their contribution and all people who are helping to manage the non-trivial DOI registration process and web presentation pages for Online SNE and layout of SNE.

Felix Breitenecker, SNE Editor-in-Chief,  
 eic@sne-journal.org; felix.breitenecker@tuwien.ac.at

### Contents SNE 22(3-4)

SNE doi: 10.11128/sne.22.1014

Calibration of a Dynamic Model of a Full Scale Wastewater Treatment Plant for Prediction of the Potential of Combined In-line Hydrolysis with Predenitrification. T. Hey, K. Jönsson, J. la Cour Jansen .....	115
Optimal Experiment Design for Calibrating an Airpath Model of a Diesel Engine. I. Stamati, D. Telen, F. Logis, E. Van Derlinden, M. Hirsch, T. Passenbrunner, J. Van Impe .....	121
Predictions of Carbon Fixation during a Bloom of <i>Emiliana huxleyi</i> as a Function of the Regulating Inorganic Species. O. Bernard, A. Sciandra, S. Rabouille .....	129
Dynamic Modeling of a Methanization Plant. A. Bader, S. Bauersfeld, C. Brunhuber, R. Pardemann, B. Meyer .....	135
Dynamic Modeling and Simulation of Low Density Polyethylene Production – A Comparative Study. I. Disli, A. Kienle .....	141
A Software Framework for Modeling and Simulation of Dynamic Metabolic and Isotopic Systems. J. Tillack, S. Noack, K. Nöh, A. Elsheikh, W. Wiechert .....	147
Advanced Optimization of Hybrid-Electric-Vehicle Drivelines using Engine-in-the-Loop Simulation. B. Tilch, T. Reimers, P. Eilts .....	157
Software for Higher-order Sensitivity Analysis of Parametric DAEs. M. Schmitz, R. Hannemann-Tams, B. Gendler, M. Förster, W. Marquardt, U. Naumann .....	163
Construction and Implementation of a Simple Agent-Based System on GPU-Architectures G. Schneckenreither, S. Hepp, D. Prokesch .....	169
Physical Modelling in MvStudium Y. Kolesov, Y. Senichenkov .....	177
BroSAnT – an Example of the Mechatronical Way of Thinking S. Petkun, X. Zhao .....	181
Training Method of Oil and Gas Industry Operators on the Example of the Automaton Simulation Usage E. Gromakov, M. Sotnikova .....	187
EUROSIM Societies Info & News .....	N1-N7

### Reader's Info

Simulation Notes Europe publishes peer reviewed *Technical Notes*, *Short Notes* and *Overview Notes* on developments and trends in modelling and simulation in various areas and in application and theory. Furthermore SNE documents the ARGESIM Benchmarks on *Modelling Approaches and Simulation Implementations* with publication of definitions, solutions and discussions (*Benchmark Notes*). Special *Educational Notes* present the use of modelling and simulation in and for education and for e-learning.

SNE is the official membership journal of EUROSIM, the Federation of European Simulation Societies. A News Section in SNE provides information for EUROSIM Simulation Societies and Simulation Groups.

SNE is published in a printed version (Print ISSN 2305-9974) and in an online version (Online ISSN 2306-0271). With Online SNE the publisher ARGESIM follows the Open Access strategy, allowing download of published contributions for free. Since 2011 Online SNE contributions are identified by a DOI (Digital Object Identifier) assigned to the publisher ARGESIM (DOI prefix 10.11128). Print SNE, high-resolution Online SNE, full SNE Archive, and source codes of the *Benchmark Notes* are available for subscription via membership in a EUROSIM society.

SNE Print ISSN 2305-9974, SNE Online ISSN 2306-0271  
 SNE Issue 22(3-4) Dec. 2012 doi: 10.11128/sne.22.1014

→ [www.sne-journal.org](http://www.sne-journal.org)

✉ [office@sne-journal.org](mailto:office@sne-journal.org), [eic@sne-journal](mailto:eic@sne-journal)

✉ SNE Editorial Office, c/o ARGESIM / DWH,  
 Neustiftgasse 57-59, 1070 Vienna, Austria

## SNE Editorial Board

SNE - Simulation Notes Europe is advised and supervised by an international scientific editorial board. This board is taking care on peer reviewing and handling of *Technical Notes*, *Education Notes*, *Short Notes*, *Software Notes*, *Overview Notes*, and of *Benchmark Notes* (definitions and solutions). At present, the board is increasing:

Felix Breitenecker, [Felix.Breitenecker@tuwien.ac.at](mailto:Felix.Breitenecker@tuwien.ac.at)

Vienna Univ. of Technology, Mathematical Modelling and Simulation; Austria, Editor-in-chief

Maja Atanasijevic-Kunc, [maja.atanasijevic@fe.uni-lj.si](mailto:maja.atanasijevic@fe.uni-lj.si)

Univ. of Ljubljana, Lab. Modelling & Control, Slovenia

Peter Breedveld, [P.C.Breedveld@el.utwente.nl](mailto:P.C.Breedveld@el.utwente.nl)

University of Twente, Div. Control Engineering, Netherlands

Agostino Bruzzone, [agostino@itim.unige.it](mailto:agostino@itim.unige.it)

Universita degli Studi di Genova, Italy

Francois Cellier, [fcellier@inf.ethz.ch](mailto:fcellier@inf.ethz.ch)

ETH Zurich, Institute for Computational Science, Switzerland

Vlatko Čerić, [vceric@efzg.hr](mailto:vceric@efzg.hr)

Univ. Zagreb, Fac. of Organization and Informatics, Croatia

Russell Cheng, [rhc@maths.soton.ac.uk](mailto:rhc@maths.soton.ac.uk)

University of Southampton, Fac. Mathematics/OR Group, UK

Horst Ecker, [Horst.Ecker@tuwien.ac.at](mailto:Horst.Ecker@tuwien.ac.at)

Vienna Univ. of Technology, Inst. f. Mechanics, Austria

Edmond Hajrizi, [ehajrizi@ubt-uni.net](mailto:ehajrizi@ubt-uni.net)

University for Business and Technology, Pristina, Kosovo

András Jávör, [javor@eik.bme.hu](mailto:javor@eik.bme.hu),

Budapest Univ. of Technology and Economics, Hungary

Esko Juuso, [esko.juuso@oulu.fi](mailto:esko.juuso@oulu.fi)

Univ. Oulu, Dept. Process/Environmental Eng., Finland

Kaj Juslin, [kaj.juslin@vtt.fi](mailto:kaj.juslin@vtt.fi)

VTT Technical Research Centre of Finland, Finland

Francesco Longo, [f.longo@unical.it](mailto:f.longo@unical.it)

Univ. of Calabria, Mechanical Department, Italy

David Murray-Smith, [d.murray-smith@elec.gla.ac.uk](mailto:d.murray-smith@elec.gla.ac.uk)

University of Glasgow, Fac. Electrical Engineering, UK

Gasper Music, [gasper.music@fe.uni-lj.si](mailto:gasper.music@fe.uni-lj.si)

Univ. of Ljubljana, Fac. Electrical Engineering, Slovenia

Thorsten Pawletta, [pawel@mb.hs-wismar.de](mailto:pawel@mb.hs-wismar.de)

Univ. Wismar, Dept. Computational Engineering,

Wismar, Germany

Niki Popper, [niki.popper@drahtwarenhandlung.at](mailto:niki.popper@drahtwarenhandlung.at)

dwh Simulation Services, Vienna, Austria

Thomas Schriber, [schriber@umich.edu](mailto:schriber@umich.edu)

University of Michigan, Business School, USA

Yuri Senichenkov, [sneyb@dcn.infos.ru](mailto:sneyb@dcn.infos.ru)

St. Petersburg Technical University, Russia

Sigrid Wenzel, [S.Wenzel@uni-kassel.de](mailto:S.Wenzel@uni-kassel.de)

University Kassel, Inst. f. Production Technique, Germany

## Author's Info

Authors are invited to submit contributions which have not been published and have not being considered for publication elsewhere to the SNE Editorial Office. SNE distinguishes different types of contributions (*Notes*):

- *Overview Note* – State-of-the-Art report in a specific area, up to 14 pages, only upon invitation
- *Technical Note* – scientific publication on specific topic in modelling and simulation, 6 – 8 (10) pages
- *Education Note* – modelling and simulation in / for education and e-learning; max. 6 pages
- *Short Note* – recent development on specific topic, max. 4 pages
- *Software Note* – specific implementation with scientific analysis, max 4 pages
- *Benchmark Note* – Solution to an ARGEIM Benchmark; basic solution 2 pages, extended and commented solution 4 pages, comparative solutions on invitation

Interested authors may find further publication and layout information at SNE's website → [www.sne-journal.org](http://www.sne-journal.org) (templates for *Notes*, requirements for benchmark solutions, etc.).

## SNE Contact & Info

→ [www.sne-journal.org](http://www.sne-journal.org)

✉ [office@sne-journal.org](mailto:office@sne-journal.org), [etc@sne-journal.org](mailto:etc@sne-journal.org)

✉ SNE Editorial Office, c/o ARGESIM / DWH, Neustiftgasse 57-59, 1070 Vienna, Austria

### SNE SIMULATION NOTES EUROPE

ISSN SNE Print ISSN 2305-9974, SNE Online ISSN 2306-0271

WEB: → [www.sne-journal.org](http://www.sne-journal.org), DOI prefix 10.11128/sne

Scope: Technical Notes, Short Notes and Overview Notes on developments and trends in modelling and simulation in various areas and in application and theory; benchmarks and benchmark documentations of ARGESIM Benchmarks on modelling approaches and simulation implementations; modelling and simulation in and for education, simulation-based e-learning; society information and membership information for EUROSIM members (Federation of European Simulation Societies and Groups).

Editor-in-Chief: Felix Breitenecker, Vienna Univ. of Technology, Inst. f. Analysis and Scientific Computing, Div., Math. Modelling and Simulation, Wiedner Hauptstrasse 8-10, 1040 Vienna, Austria;

✉ [Felix.Breitenecker@tuwien.ac.at](mailto:Felix.Breitenecker@tuwien.ac.at), ✉ [etc@sne-journal.org](mailto:etc@sne-journal.org)

Layout / Administration: Anna Mathe, ARGESIM / ASIM

✉ [anna.mathe@tuwien.ac.at](mailto:anna.mathe@tuwien.ac.at), ✉ [office@sne-journal.org](mailto:office@sne-journal.org)

Print SNE: Grafisches Zentrum, TU Vienna,

Wiedner Hauptstrasse 8-10, 1040, Vienna, Austria

Online SNE: DWH Simulation Services, Neustiftgasse 57-59, 1070 Vienna

Publisher: ARGESIM ARBEITSGEMEINSCHAFT SIMULATION NEWS

- WORKING COMMITTEE SIMULATION NEWS, Neustiftgasse 57-59,

1070 Vienna, Austria; → [www.argesim.org](http://www.argesim.org), ✉ [info@argesim.org](mailto:info@argesim.org)

on behalf of ASIM(→ [www.asim-gi.org](http://www.asim-gi.org) and EUROSIM

→ [www.eurosim.info](http://www.eurosim.info)

© ARGESIM / EUROSIM / ASIM 2012

# Calibration of a Dynamic Model of a Full Scale Wastewater Treatment Plant for Prediction of the Potential of Combined In-line Hydrolysis with Predenitrification

Tobias Hey<sup>1\*</sup>, Karin Jönsson<sup>2</sup>, Jes la Cour Jansen<sup>2</sup>

<sup>1</sup> VA SYD, P.O. Box 191, 201 21 Malmö, Sweden; \* [tobias.hey@vasyd.se](mailto:tobias.hey@vasyd.se)

<sup>2</sup> Water and Environmental Engineering, Dept. Chemical Engineering, Univ. Lund, Sweden

**Abstract.** Combining sludge hydrolysis with predenitrification is a promising method to remove nitrogen and decreased use of external carbon content in wastewater. The Klagshamn wastewater treatment plant plans to alter the primary settler into an in-line hydrolysis basin with subsequent predenitrification to reduce the nitrate load into the overstrained moving-bed biofilm reactor. A reference scenario, which included annual data from the process control system, was established and calibrated with the operational findings from the plant in 2007 to form a dynamic wastewater model. A method for generating a complete annual data set based on laboratory values was established and implemented with the model. The simulated results agreed well with Klagshamn's actual operation. In addition, the simulated results of the proposed in-line hydrolysis basin were used to evaluate the potential for predenitrification. The results with hydrolysate showed that an additional  $40 \text{ t}\cdot\text{a}^{-1}$  of nitrogen could be denitrified compared to the reference scenario.

## Introduction – SNE Header Unnumb.

The Klagshamn municipal wastewater treatment plant (WWTP) in Sweden uses mechanical and chemical treatment as the primary treatment step. Thereafter, carbon is removed and ammonium is converted into nitrate in the activated sludge (AS) process. Nitrogen removal occurs in a subsequent moving-bed biofilm reactor (MBBR<sup>TM</sup>) with ethanol as an external carbon source functioning as an energy supply for the denitrifying heterotrophic bacteria (Henze et al., 2002).

In 1998, a study predicted that the Klagshamn WWTP would reach its maximum capacity for nitrogen within 15 years due to the increasing number of connected households (Andersson et al., 1998).

A possible extension of the MBBR<sup>TM</sup> volume or carrier-filling degree was determined to be too costly. Converting the WWTP into a chemical-free plant that could provide the required capacity is a major challenge. Therefore, Jönsson et al. (2008) performed a study investigating the predenitrification potential by applying primary sludge hydrolysis to provide the internal carbon source. A denitrification rate of  $3.1 \text{ mg NO}_3\text{-N}\cdot(\text{g VSS}\cdot\text{h})^{-1}$  for the hydrolysate was achieved, and 50% of the external carbon source could be saved. To reduce the nitrate load into the MBBR<sup>TM</sup>, between 12.5% and 25% of the AS volume could be used for predenitrification. To determine the corresponding values in a full-scale application, dynamic variations such as flow, temperature and wastewater compound concentrations needed to be included in the dynamic wastewater modelling to evaluate possible scenarios for the Klagshamn WWTP. Dynamic wastewater modelling was able to effectively evaluate the AS process on other full-scale WWTPs (Brdjanovic et al., 2000; Larrea et al., 2000; Makinia et al., 2006). However, due to cost and time limitations, the characterisation of the wastewater compositions and variations are less satisfactory when modelling over a full year of operation.

In addition, the Swedish Environmental Protection Agency (NFS; 2000) stipulates that wastewater should be sampled on a 24 h flow-proportional basis, resulting in a limited number of samples that can be used for plant control. Therefore, the amount of analysed wastewater compounds (e.g. total chemical oxygen demand (COD<sub>t</sub>), total nitrogen (TN) and total phosphorous (TP<sub>t</sub>)) can vary and lead to limited data on some of the main wastewater components. A method for generating reasonable estimates of the missing wastewater compound concentrations over the course of a year needs to be established.

This study presents a dynamic simulation containing configurations such as tank volumes, tank areas and pump capacities of the Klagshamn WWTP from 2007, which includes the year-round data for flow and temperature variations applied in the wastewater simulation tool EFOR (2003). A method for establishing a complete annual time series of the incoming wastewater composition based on measured laboratory values is presented. The model is then calibrated to reflect the actual performance of Klagshamn from 2007 denoted as the reference scenario. The composition of the AS that results from the introduction of inline hydrolysis is used with the model to assess the potential for pre-denitrification.

### 1 Model Presentation

The full-scale properties of Klagshamn WWTP are presented in Table 1 and were applied to the wastewater treatment simulation tool EFOR 2003.0 based on the Activated Sludge Model (ASM) 2d (Henze *et al.*, 1999).

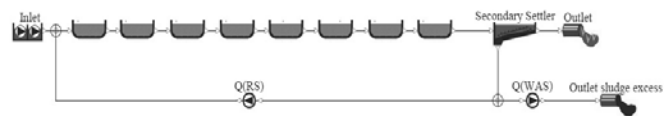
Name	Volume	Depth	Flow rate
AS tank Bot- tom	8·550 m <sup>3</sup>	4 m	1000 kgO <sub>2</sub> ·h <sup>-1</sup>
Secondary settler	8·610 m <sup>3</sup>	3.6 m	
Return sludge (Q(RS))			1000 m <sup>3</sup> ·h <sup>-1</sup>
Waste sludge (Q(WAS))			30 m <sup>3</sup> ·h <sup>-1</sup>

**Table 1.** Analysis methods and number of measurements from the outlet of the primary settler during one operational year.

To simulate operation strategies with different wastewater characteristics (e.g., increased volatile fatty acid (VFA) concentration due to in-line primary sludge hydrolysis), the modelling framework was determined in accordance with Klagshamn’s treatment steps, as presented in Figure 1.

The outlet of either the primary settler or primary sludge hydrolysis tank was used as the inlet into the AS system. In the model, the eight AS tanks in series represent eight AS zones that can be operated in either oxic or anoxic conditions (Nyberg *et al.*, 1992).

The secondary settler’s sludge separation performance based on the model of Takacs *et al.* (1991) was connected with the Q(RS) and Q(WAS) pump for sludge control in the AS system. For modelling reasons, the areas of the eight secondary settlers were summed up, and one secondary settler with an area of 4880 m<sup>2</sup> was used instead. To control the aeration in the AS tanks, Q(RS) and Q(WAS) control loops were set up, as presented in Table 2.



**Figure 1.** Layout of the AS system including the secondary settler, the return AS pump and waste AS pump in the dynamic simulation tool.

Controlled Item	Meter	Controller	Set-point	Units
Aerators AS1-8	Oxygen AS1-8	On/Off	2.5	g·m <sup>-3</sup>
Q(RS)	Time	Time	1000	m <sup>3</sup> ·h <sup>-1</sup>
Q(WAS)	SS <sub>AS</sub> in AS8	On/Off	30	m <sup>3</sup> ·h <sup>-1</sup>

**Table 2.** Set-up of the control loops in the dynamic simulation tool.

#### 1.1 Data Background

The number of analyses (n) and corresponding method of measurement at the outlet of the primary settler at Klagshamn during 2007 are shown in Table 3. The analysed compounds were suspended solids (SS), COD<sub>t</sub>, TN, ammonium nitrogen (NH<sub>4</sub>-N), TP<sub>t</sub>, filtered phosphorus (TPs) and the sludge concentrations in the AS tank (SS<sub>AS</sub>). Furthermore, the incoming wastewater flow (Q<sub>in</sub>) and wastewater temperature (T<sub>in</sub>) was obtained using the supervisory control and data acquisition (SCADA) system.

	Unit	Annual Measurement		Method
		n	%	
SS	gSS·m <sup>-3</sup>	82	22	SS-EN 872
CODt	gO <sub>2</sub> ·m <sup>-3</sup>	303	83	LCK 114
TN	gN·m <sup>-3</sup>	311	85	FIA 62-04/84
NH <sub>4</sub> -N	gN·m <sup>-3</sup>	70	19	FIA 50-02/84
TPt	gP·m <sup>-3</sup>	236	65	SS 028127-2
TPs	gP·m <sup>-3</sup>	215	59	SS 028127-2
SS <sub>AS</sub>	gSS·m <sup>-3</sup>	109	30	SS-EN 872
Q <sub>in</sub>	m <sup>3</sup> ·d <sup>-1</sup>	365	100	SCADA
T <sub>in</sub>	°C	365	100	SCADA

**Table 3.** Analysis methods and number of measurements from the outlet of the primary settler during one operational year.

## 2 RESULTS

### 2.1 Modelling Strategy

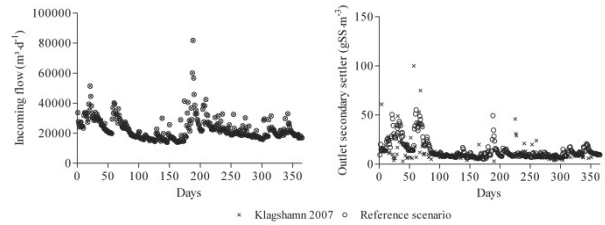
To reflect the measured data (including hydraulic variations), the model was empirically calibrated to match the actual SS measurements in the AS and the secondary settler. The wastewater treatment plant properties, process adjustments, and the wastewater treatment model that was implemented described as the reference scenario.

All data from Q<sub>in</sub> were incorporated directly into the model. Two typical operational situations are shown in Figure 2. The flow is either normal (Q<sub>d, avg</sub> ~ 23 000 m<sup>3</sup>·d<sup>-1</sup>) or very high (Q<sub>d, max</sub> > 39 000 m<sup>3</sup>·d<sup>-1</sup>) due to snowmelt in the winter/spring (day 21 and 60) or heavy rainfall in the summer (day 187), respectively.

### 2.2 Secondary Settler

The number of layers in the flux settler (n<sub>L</sub>) and height (m<sub>L</sub>) were changed from n<sub>L</sub> = 36 and 0.1 m<sub>L</sub> (default) to n<sub>L</sub>=10 and 0.36 m<sub>L</sub>, respectively. To achieve a stable sludge concentration in the AS system while avoiding sludge loss, the default value of the non-settling fractions (f<sub>ns</sub>) of 0.23% was set to zero.

Figure 2 shows the SS concentrations in the reference scenario and the actual measured values from the outlet of the secondary settler at Klagshamn during 2007.



**Figure 2.** Q<sub>in</sub> and SS from outlet secondary settler at Klagshamn WWTP during 2007. Day 0 represents Jan. 1, 2007.

### 2.3 Activated Sludge System

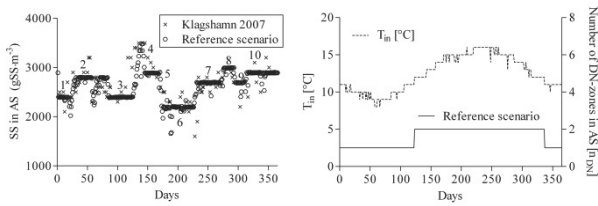
All SS measurements from the AS system at Klagshamn during 2007 were divided into ten periods with fixed set-points, as presented in Table 4. These set-points were chosen by looking at the actual SS concentrations, which were almost constant for longer periods in the AS tank (see Figure 3).

Phase	Calendar days	Duration (days)	Klagshamn 2007 (gSS·m <sup>-3</sup> )	Set-point in EFOR (gSS·m <sup>-3</sup> )
1	001 – 022	22	2425	2400
2	023 – 082	60	2691	2800
1	001 – 022	22	2425	2400
2	023 – 082	60	2691	2800
3	083 – 124	42	2633	2400
4	125 – 141	17	3450	3500
5	142 – 169	28	3057	2900
6	170 – 225	56	2305	2200
7	226 – 270	45	2568	2700
8	271 – 292	22	2871	3000
9	293 – 309	17	2533	2700
10	310 – 365	56	2844	2900

**Table 4.** Mean values of SS concentration in the AS system and the SS set-points in the dynamic simulation at Klagshamn during 2007.

The actual number of zones operated under anoxic conditions (n<sub>DN</sub>) in the AS system and T<sub>in</sub> were included into the model presented in Figure 3. The maximal growth rate (μ<sub>A</sub>) for the autotrophic biomass was changed from 0.9 d<sup>-1</sup> (default) to 1.25 d<sup>-1</sup>, and the fraction of denitrifiers in the AS system was changed from 0.6 (default) to 1.0 to calibrate the model of the AS process empirically.





**Figure 3.** (left) Actual sludge concentration in the AS tank at Klagshamn during 2007 and in the reference scenario. (right) Number of anoxic zones in the AS tank in the reference scenario as a function of the inlet temperature.

### 2.4 Completing the Annual Composition and Variation of the Inflow Data

Extensive data were required to accurately perform an annual dynamic simulation with 24 h time steps. At Klagshamn, sampling and analyses of wastewater compounds were not performed on a daily basis, as noted in Table 3. Moreover, not all compounds were analysed simultaneously, resulting in different numbers (n) of measurements (e.g., n = 303 for CODt and n = 82 for SS, as presented in Figure 4).

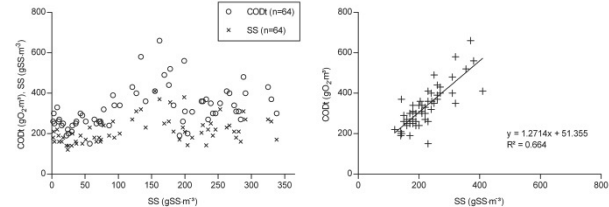
To characterise the incoming wastewater and find any potential relationship between the measured compounds, a search of the literature was performed. According to Henze *et al.* (2002, 2008), some compounds appear in municipal wastewater in typical ratios, as presented in Table 5. Therefore, the ratio of CODt/SS, NH<sub>4</sub>-N (S<sub>NH4</sub>)/TN and TP<sub>s</sub>/TP<sub>t</sub> were computed to compare to literature values.

Compounds	Pairs	Ratio (raw)	R <sup>2</sup>	Ratio (processed)	Henze <i>et al.</i> (2002, 2008)
CODt/SS	64	1.5	0.664	1.5	1.4 - 1.6
NH <sub>4</sub> -N/TN	61	0.7	0.878	0.7	0.6
TP <sub>s</sub> /TP <sub>t</sub>	215	0.5	0.696	0.5	0.5

**Table 5.** Comparison of raw and processed compound ratios from this study with literature.

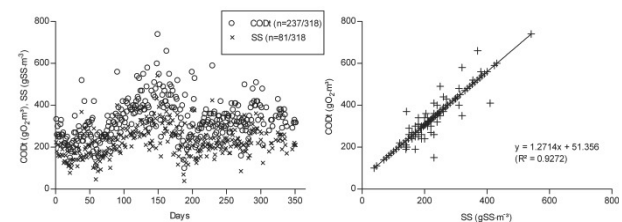
Linear regression (De Veaux *et al.*, 2008) was performed to estimate the missing values in the incomplete data set; CODt and SS estimates are presented below as examples. First, all CODt and SS concentrations measured during the same day were selected for a total of 64 pairs. Subsequently, a mean CODt/SS ratio of 1.5 was calculated and compared to the expected ratio (Table 5).

As shown in Figure 4, a scatter plot of SS (x-axis) versus CODt (y-axis) is presented to visualise the linear equation ( $y = 1.2714 \cdot x + 51.355$ ,  $y = \text{CODt}$ ,  $x = \text{SS}$ ) and goodness-of-fit ( $R^2 = 0.664$ ).



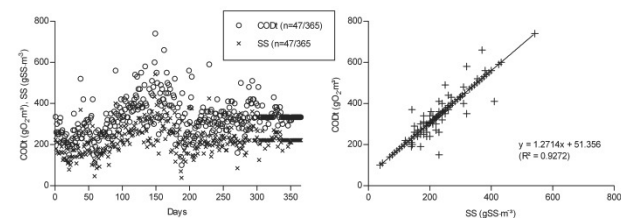
**Figure 4.** (left) Time series of CODt and SS pairs. (right) Scatter plot with the line of best fit.

Next, the linear equation was used to calculate the remaining CODt (n = 81) and SS (n = 237) values that were not measured on the corresponding day. A time series and a scatter plot with a line of best fit that included all calculated CODt and SS values were made (Figure 5).



**Figure 5.** (left) Time series and (right) scatter plot with the best-fit line that include all calculated CODt and SS values.

On days when neither the CODt nor the SS values were measured, the annual average values (334 gCODt·m<sup>-3</sup> and 222 gSS·m<sup>-3</sup>) from Figure 5 were used instead. Figure 6 presents a complete annual data set (n = 365) for CODt and SS values including the additional annual average values (n = 47).



**Figure 6.** Complete annual data set of (left) CODt and SS values. (right) Corresponding scatter plot.



This method was used to estimate the missing data values for TPt, TPs, TN and  $\text{NH}_4\text{-N}$ .

## 2.5 Wastewater Characterisation

To characterise the wastewater in more detail, the *weak primary settled wastewater* was chosen as the default wastewater type based on the wastewater compounds analysed (SS, CODt, TN,  $\text{NH}_4\text{-N}$ , TPt and TPs). This setting uses predefined ratios of compounds typical of this kind of wastewater when no data are available (Henze *et al.*, 2002).

The fraction of soluble inert COD ( $f_{\text{S,SI}}$ ) in the total COD was calculated based on laboratory analyses using the method provided by Keskitalo *et al.* (2010). Therefore, the  $f_{\text{S,SI}}$  was changed from 0.03  $\text{gCOD}\cdot\text{gCOD}^{-1}$  (default) to 0.12  $\text{gCOD}\cdot\text{gCOD}^{-1}$ .

## 2.6 Implementing Sludge Hydrolysis to Characterise Incoming Wastewater

Estimates of primary sludge hydrolysis were generated for use in full-scale simulations by changing the reference scenario's wastewater characteristics. First, the VFA produced were set to increase simultaneously with the CODt and soluble COD (CODs) in the model (Henze *et al.*, 2002, 2008; Peña-Roja *et al.*, 2002). Therefore, the VFA was added to the CODt and CODs while the default value of the VFA ( $S_A$ ) was changed.

Furthermore, any release of  $\text{NH}_4\text{-N}$  ( $S_{\text{NH}_4}$ ) or ortho-phosphate ( $S_{\text{PO}_4}$ ) would be added to the total nitrogen content ( $C_{\text{TN}} = S_{\text{NO}_X} + S_{\text{NH}_4} + \text{organic N}$ ) or total phosphorous ( $C_{\text{TP}} = S_{\text{PO}_4} + S_{\text{p-P}} + \text{organic P}$ ), respectively (Henze *et al.*, 2002).

## 2.7 Simulation of Sludge Hydrolysis Results for Improved Predenitrification

Hey *et al.* presents a full-scale in-line hydrolysis experiment at Klagshamn, and a value of  $43 \text{ gCO-D}_{\text{HAC}}\cdot\text{m}^{-3}$  was noted with no ammonium release. These findings were incorporated into the model and simulated with the reference scenario. The simulated results in Figure 7 show that an additional load of  $40 \text{ tNO}_3\text{-N}\cdot\text{a}^{-1}$  could be removed, corresponding to an annual average concentration of less than  $4.7 \text{ gNO}_3\text{-N}\cdot\text{m}^{-3}$  in the hydrolysate.

However, a load of  $15 \text{ tNH}_4\text{-N}\cdot\text{a}^{-1}$  less was nitrified, corresponding to an excess of  $1.6 \text{ gNH}_4\text{-N}\cdot\text{m}^{-3}$  annual average outlet concentration that is supplied to the downstream process.

Nevertheless, the simulation indicated a potential for removing a total nitrogen load of  $25 \text{ tN}\cdot\text{a}^{-1}$ , which corresponds to  $3.1 \text{ gN}\cdot\text{m}^{-3}$  less into the downstream process.

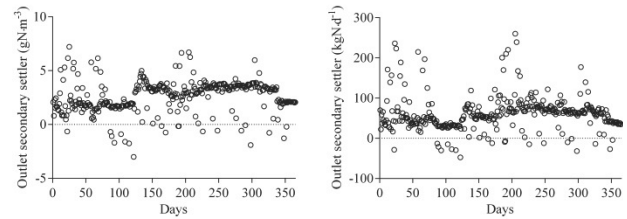


Figure 7. Simulated outlet concentrations of (left) ammonium nitrogen and (right) nitrate nitrogen from the secondary settler.

## 3 Discussion

Generating a complete annual data set for modelling purposes by applying linear regression was shown to produce estimates in good agreement with available data. Furthermore, the data sets agreed well with values related to typical inlet wastewater compound ratios, as reported in the literature (Henze *et al.*, 2002, 2008). The reasonably large number of annual measurements of TN ( $n = 311$ ; 85%), CODt ( $n = 303$ ; 83%), TPt ( $n = 236$ ; 65%) and TPs ( $n = 215$ ; 59%) enabled the daily values for the more sparsely sampled  $\text{NH}_4\text{-N}$  ( $n = 70$ ; 19%) and SS ( $n = 82$ ; 22%) to be estimated.

The main purpose was to attain a complete data set for TN and TPt supported by experimental measurements and estimations based on the ratio of other wastewater compounds. If the wastewater compositions were incorrectly estimated, the daily load in the wastewater simulation tool would also be incorrect, resulting in false evaluation and conclusions. Almost no differences were found when the SS load from the outlet of the secondary settler between the reference scenario was compared to the actual findings from 2007 (Figure 2). This result indicates that the empirical calibration fits the data accurately.

The reference scenario was shown to be an appropriate basis for the simulation of different future scenarios. For example, the inlet wastewater characteristics could be changed to predict the potential of combining in-line hydrolysis with predenitrification to remove nitrogen in the AS process.

## 4 Conclusion

Dynamic wastewater treatment simulation was applied in this study and shown to be a valuable tool to quickly evaluate the potential operational changes without the risk associated with full-scale testing. Routine measurements made at most plants are not performed with the time resolution desired for modelling purposes. However, with proper treatment, less frequent measurements could be made useful. This study shows a method for estimating missing wastewater compound concentrations to obtain a complete annual data set that can be implemented in a calibrated dynamic wastewater treatment simulation tool to evaluate primary sludge hydrolysis for full-scale predenitrification.

### References

- [1] Andersson, B., Aspegren, H., Nyberg, U., Jansen, J.L.C. and Ødegaard, H. (1998). Increasing the capacity of an extended nutrient removal plant by using different techniques. *Water Science and Technology*, 37, 63-71.
- [2] Brdjanovic, D., van Loosdrecht, M.C.M., Versteeg, P., Hooijmans, C.M., Alaerts, G.J. and Heijnen, J.J. (2000). Modeling COD, N and P removal in a full-scale wwtp Haarlem Waarderpolder. *Water Research*, 34, 846-858.
- [3] De Veaux, R.D., Velleman, P.F. and Bock, D.E. (2008). *Stats: data and models*. Addison-Wesley, Boston.
- [4] EFOR 2003.0. (2003). DHI Hørsholm, Hørsholm, Denmark; software available [www.dhisoftware.com/efor](http://www.dhisoftware.com/efor).
- [5] Henze, M., Gujer, W., Mino, T., Matsuo, T., Wentzel, M.C., Marais, G.V.R. and van Loosdrecht, M.C.M. (1999). Activated sludge model No. 2d, ASM2d. *Water Science and Technology*, 39, 165-182.
- [6] Henze, M., Harremoës, P., Jansen, J.L.C. and Arvin, E. (2002). *Wastewater treatment: biological and chemical processes*. Third edition. Springer, Berlin.
- [7] Henze, M., van Loosdrecht, M.C.M., Ekama, G.A. and Brdjanovic, D. (2008). *Biological wastewater treatment: principles, modelling and design*. IWA Publication, London.
- [8] Hey, T., Jönsson, K. and Jansen, J.L.C. Full scale in-line hydrolysis and simulation for potential energy and resource savings in activated sludge – a case study. Accepted for publication in *Environmental Technology*.
- [9] Jönsson, K., Pottier, A., Dimitrova, I. and U. Nyberg. (2008). Utilising laboratory experiments as a first step to introduce primary sludge hydrolysis in full-scale. *Water Science and Technology*, 57, 1397-1403.
- [10] Keskitalo, J., Jansen, J.L.C. and Leiviskä, K. (2010). Calibration and validation of a modified ASM1 using long-term simulation of a full-scale pulp mill wastewater treatment plant. *Environmental Technology*, 31, 555-566.
- [11] Larrea, L., Irizar, I. and Hidalgo, M.E. (2000). Improving the predictions of ASM2d through modelling in practice. *Water Science and Technology*, 45, 199-208.
- [12] Makinia, J., Rosenwinkel, K.H. and Spering, V. (2006). Comparison of two model concepts for simulation of nitrogen removal at a full-scale biological nutrient removal pilot plant. *Journal of Environmental Engineering – ASCE*, 132, 476-487.
- [13] Nyberg, U., Aspegren, H., Andersson, B., Jansen, J.L.C. and Villadsen, I.S. (1992). Full-scale application of nitrogen removal with methanol as carbon source. *Water Science and Technology*, 26, 1077-1086.
- [14] Penya-Roja, J.M., Seco, A., Ferrer, J. and Serralta, J. (2002). Calibration and validation of activated sludge model No.2d for Spanish municipal wastewater. *Environmental Technology*, 23, 849-862.
- [15] Swedish Environmental Protection Agency. (2000). *Naturvårdsverkets författningssamling NFS 2000:15* (In Swedish).
- [16] Takacs, I., Patry, G.G. and Nolasco, D. (1991). A dynamic model of the clarification-thickening process. *Water Research*, 25, 1263-1271.

Submitted MATHMOD 2012: November 2011

Accepted MATHMOD 2012: December 2012

Submitted SNE Revised: October 2012

Accepted: December 5, 2012

# Optimal Experiment Design for Calibrating an Airpath Model of a Diesel Engine

Ioanna Stamati<sup>1</sup>, Dries Telen<sup>1</sup>, Filip Logist<sup>1</sup>, Eva Van Derlinden<sup>1</sup>, Markus Hirsch<sup>2</sup>,  
Thomas Passenbrunner<sup>2</sup>, Jan Van Impe<sup>1\*</sup>

<sup>1</sup> BioTeC & OPTEC, Chemical Engineering Department, Katholieke Universiteit Leuven, W. de Croylaan 46, 3001 Leuven, Belgium; \*[jan.vanimpe@cit.kuleuven.be](mailto:jan.vanimpe@cit.kuleuven.be)

<sup>2</sup> Institute for Design and Control of Mechatronical Systems, Johannes Kepler University, Linz, Austria

**Abstract.** Mathematical models become more and more indispensable tools for engine manufacturers. As nonlinear dynamic models based on first-principles are preferred by practitioners, model calibration or parameter estimation is often a time consuming task. The use of optimally designed dynamic inputs can reduce the experimental burden and increase the accuracy of the estimated parameters. The current paper presents the calibration and validation of a Diesel engine airpath model. Optimal inputs have been designed based on random phase multisine inputs. These multisines can be adapted to excite exclusively a specific frequency band of interest. Moreover, they allow (i) to concentrate the input around an operating point, and (ii) to include fast variations in the input profile without introducing a large number of discretization parameters. The resulting model has been found to provide an acceptable predictive power in both identification and validation.

## Introduction

Mathematical models and simulations are more and more exploited for the analysis, design, operation and optimization of engines (Stewart et al., 2011). However, the accurate modelling of engine processes is often a non-trivial task. There are several reasons for this. First, many of these processes are intrinsically *dynamic* in nature, i.e., properties and variables vary over time, giving rise to dynamic models. As often fast and slowly varying variables are present, different time scales have to be accounted for. Moreover, *mechanistic* models, which start from the physical principles and conservation laws underlying the process, are preferred in practice because of their generic prediction capabilities.

However, the underlying mechanisms often require highly nonlinear model descriptions. Mathematically,

this kind of description results in non-linear ordinary differential equation (ODE) models. In addition, before the models can be employed in practice for the above mentioned purposes, *model calibration* or *parameter estimation (PE)* is required, i.e., estimating the unknown parameters based on experimental data. As experiments can be time consuming and expensive, Optimal Experiment Design (OED) techniques are attractive to reduce the experimental burden (Walter and Pronzato, 1997). Here, optimal inputs are designed such that outputs are as informative as possible with respect to the target parameter(s).

In the current paper dynamic parameters of an airpath model for a Diesel engine have to be estimated. The airpath model used is similar to the one described in Puchner et al. (2009) and the dynamic parameters have to be estimated based on measurement from an engine test bench.

In the current case, two specific requirements are present. Fast variations in the designed inputs have to be combined with long time horizons and (too) large deviations from specified setpoints have to be avoided. This combination prohibits the use of polynomial discretizations which are typically employed (Franceschini and Macchietto, 2008). Hence, procedures based on multisine input representations have been developed and successfully been applied.

The paper is organized as follows. Section 1 details the Engine test bench and the airpath model under study. In Section 2 the Optimal Experiment Design strategy is outlined whereas in Section 3 the overall approach is pointed out. Section 4 discusses the obtained results. Finally, the conclusions are summarized in Section 5.

# 1 Engine Test Bench and Airpath Model

## 1.1 Engine Test Bench

The device under investigation is a 2 liter EU5 common rail Diesel engine with external exhaust gas recirculation and a variable geometry turbine turbocharger. This new engine replaces the BMW M47T-OL Diesel engine used in Alberer (2009); Ferreau *et al.* (2007); Puchner *et al.* (2009)).

Figure 1 presents a schematic view of the engine's air path. It consists of the intake and exhaust manifold, a path for the Engine Gas Recirculation (EGR) (i.e., EGR-valve and EGR-cooler), the Variable Geometry Turbocharger (VGT) with the intercooler and the cylinder block with the swirl flaps. Five variables can be manipulated, i.e., position of EGR and VGT valve, injected fuel mass, engine speed and swirl state, while measured variables involve the intake pressure (MAP), fresh air flow (MAF), the oxygen concentration in the exhaust, the oxygen concentration in the intake manifold and the crankshaft torque.

## 1.2 Airpath model

The model is characterized by the combination of physical equations with static maps. The maps mainly capture the static behavior of the engine, while physical equations (e.g., the ideal gas equation and equations for conservation of mass and energy) account for the dynamic behavior. These maps are derived based on steady-state test-bench measurements and are used for polynomial interpolation- extrapolation between process variables (Alberer, 2009).

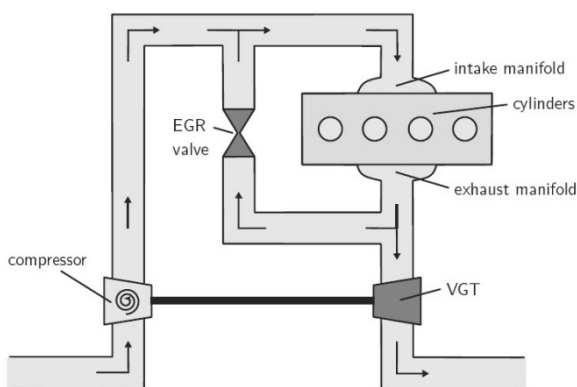


Figure 1. Schematic diagram of the Diesel airpath System (Ferreau *et al.*, 2007).

The airpath model is provided as a Simulink file which can be run from Matlab. It has as inputs  $\mathbf{u}$  the five manipulated variables, i.e., EGR valve position  $X_{egr}$ , the VGT activation signal  $X_{vgt}$ , the engine speed  $n$ , the injected fuel mass  $W_f$  and the swirl valve position. The outputs are the five mentioned measured variables, i.e., the intake pressure (MAP), fresh air flow (MAF), the oxygen concentration in the exhaust, the oxygen concentration in the intake manifold and the crankshaft torque. However, for practical reasons not all inputs and outputs have been incorporated in the Optimal Experiment Design. For instance, the swirl valve which can only be either entirely open or closed, has been assumed to be constantly open. In addition, only the in practice more easily measurable intake manifold pressure and flow have been considered to be available.

The model contains five parameters describing the dynamic behavior, which have to be estimated (i.e., *Parameter Estimation* task (PE)). These parameters are the volume of the intake and exhaust manifold  $V_i$

and  $V_x$ , the turbocharger efficiency  $\eta_n$  and the time constants of the turbocharger and the exhaust gas recirculation cooler  $\tau_{vgt}$  and  $\tau_{egr}$ . Hence, the four model inputs  $X_{egr}$ ,  $X_{vgt}$ ,  $n$  and  $W_f$  have to be designed in order to yield as informative outputs for MAF and MAP as possible (i.e., *Optimal Experiment Design* task (OED)). These two outputs mentioned above will be used for model calibration.

## 2 Optimal Experiment Design for Non-linear Dynamic Systems

### 2.1 Optimal Control Problem

Optimal Experiment Design for nonlinear dynamic systems described by Ordinary Differential Equations gives rise to a particular class of optimal control problems.

$$\min_{\mathbf{u}} J \quad (1)$$

subject to

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{p}, t) \quad t \in [0, t_f] \quad (2)$$

$$\mathbf{0} = \mathbf{b}_c(\mathbf{x}(0), \mathbf{p}) \quad (3)$$

$$\mathbf{0} \geq \mathbf{c}_p(\mathbf{x}(t), \mathbf{u}(t), \mathbf{p}, t) \quad (4)$$

Here,  $\mathbf{x}$  are the state variables,  $\mathbf{u}$  the time-varying control inputs and  $\mathbf{p}$  the model parameters.

The vector  $\mathbf{f}$  represents the dynamic system equations (on the interval  $t \in [0, t_f]$ ) with initial conditions given by the vector  $\mathbf{b}_e$ . The vector  $\mathbf{c}_p$  indicates path inequality constraints on states and controls.  $\mathbf{y}$  are the measured outputs, which are typically a subset of the state variables  $\mathbf{x}$ .

## 2.2 Objective Function

In OED, the objective function  $J$  is typically a scalar function  $\Phi$  of the Fisher information matrix

$$\mathbf{F}(\mathbf{p}) = \sum_{i=1}^{n_t} \left( \frac{\partial \mathbf{y}(\mathbf{p}, t_i)}{\partial \mathbf{p}} \right)^T \mathbf{Q} \left( \frac{\partial \mathbf{y}(\mathbf{p}, t_i)}{\partial \mathbf{p}} \right) \quad (5)$$

with  $n_t$  the number of measurement times  $t_i$ .  $\mathbf{F}$  combines information on (i) the error on the output measurements ( $\mathbf{Q}$  is typically defined as the inverse of the measurement error variance matrix), and (ii) the sensitivities of the model output ( $\mathbf{y}(\mathbf{p}, t_i)$ ) to small variations in the model parameters  $\mathbf{p}$  (expressed in the sensitivity matrix  $\frac{\partial \mathbf{y}(\mathbf{p}, t_i)}{\partial \mathbf{p}}$ ). To this end several scalar criteria have been described in literature, e.g.:

**A-criterion  $\min[\text{trace}(\mathbf{F}^{-1})]$ .** A-optimal designs minimize the arithmetic mean of the parameter estimation errors. This corresponds to the minimization of the sum of the squared axes of the asymptotic joint confidence region, i.e., minimizing the frame enclosing this confidence region.

**D-criterion  $\max[\det(\mathbf{F})]$ .** The D-criterion minimizes the geometric mean of the parameter estimation errors. D-optimal design aims at the minimization of the parameter estimation variance-covariance, i.e., minimization of the joint confidence region on  $\mathbf{p}$  via the maximization of the determinant of  $\mathbf{F}$ .

**E-criterion  $\max[\lambda_{\min}(\mathbf{F})]$ .** E-optimality focuses on the minimization of the largest parameter error (i.e., maximization of the smallest eigenvalue), and as such, neglects uncertainty on the remaining parameters. This corresponds to minimizing the longest axis of the joint confidence region

## 2.3 Input Discretization and Degrees of Freedom

Nowadays optimal control problems are most often solved by direct approaches which convert the infinite dimensional optimal control problem into a finite dimensional nonlinear program by discretizing the control resulting in a finite number of degrees to be optimized.

Typically, piecewise polynomial discretizations are used (Biegler, 2007; Diehl *et al.*, 2002). As in mechatronic systems fast variations have to be combined with long time horizons, piecewise constant control discretizations yield a too high number of control parameters to be optimized. To tackle this issue, in the current paper three strategies based on *random phase multisines* are proposed (Pintelon and Schoukens, 2001)

$$u(t) = u_0 + \sum_{k=1}^F (A_k \cos(2\pi f_k t + \varphi_k)) \quad (6)$$

where  $u_0$  is the operating point to be chosen,  $f_k$  are the frequencies and  $A_k$  are the cosine amplitudes,  $F$  is the number of frequency domain data samples and  $\varphi_k$  is a random phase. This kind of signals has the advantage that (i) energy can be concentrated in the dominant frequency ranges of the system and that (ii) they are most interesting when designed inputs are preferred around pre-specified operating points.

## 2.4 Optimization Strategies

In general, multisines exhibit three classes of degrees of freedom that can be optimized. The *root mean square* value (RMS) of the signal belongs to the first class. For a signal  $u(t)$  with  $n_t$  points, the RMS is defined as:

$$\text{RMS} = \sqrt{\frac{1}{n} \sum u^2(t_i)} \quad (7)$$

The RMS value is calculated based on the signal after subtraction of the operating point. The frequency bands compose the second class of degrees of freedom, while the last class of degrees of freedom involves the amplitudes  $A_k$  of the cosines. Depending on the selected class of degrees of freedom, different strategies are obtained. For a schematic representation see Figure 2.

In the first strategy (S1 or *RMS optimized*) only the RMS value of the multisine is optimized. This means that both the frequency band of the signal as well as the amplitudes of the different frequency contributions remain at pre-specified values.

Optimizing the frequency bands is the second strategy (S2 or *frequency optimized*). The advantage is that a frequency band can be selected that excites the outputs the most. Consequently, more information can be obtained by exciting a specific frequency band. During this optimization the RMS value is kept constant and the amplitude remains the same in the whole band.

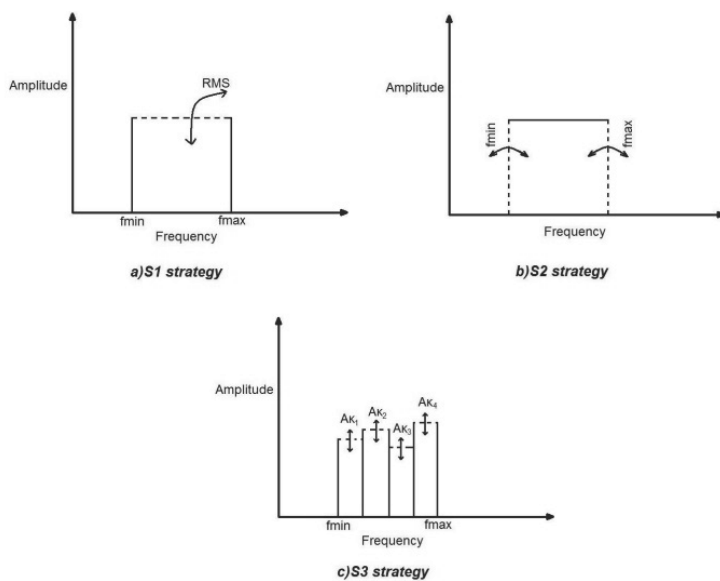


Figure 2. Schematic diagram of the different input structures.

The third strategy (S3 or *amplitude optimized*) requires to vary the amplitudes of the different frequencies. For every frequency band, the band is divided in four different parts which are allowed to have a different amplitude.

Four different subbands for each signal are chosen in order not to increase the number of decision variables too much and by doing so to limit the increase in the computation time. The amplitudes are optimized together with the RMS value.

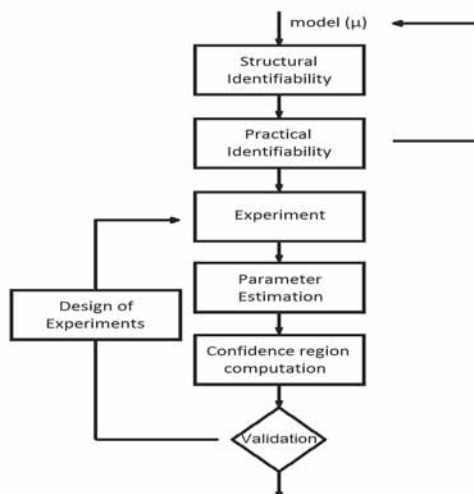


Figure 3. Schematic diagram of the modeling cycle.

### 3 Procedure

This section outlines the global approach that has been used. Under the assumption that a correct model structure is selected for the underlying dynamic process, the model parameters need to be estimated.

This can be done according to the general modeling procedure or so-called *modeling cycle* (Balsa-Canto *et al.*, 2010; Franceschini and Macchietto, 2008; Ljung, 1999; Walter and Pronzato, 1997).

The preliminary step in the *modeling cycle* involves a verification of the model's structural and practical identifiability. A sensitivity analysis of the outputs with respect to the inputs will in this respect often reveal quite some information. If this step succeeds, then a sequential approach follows for choosing the appropriate parameters as can be seen in Figure 3. This sequential part includes the loop:

*Experiment, Parameter Estimation, Confidence Interval Computation and Design of Experiments* according to the results of the Confidence Interval. This sequential procedure will be applied in the current study.

#### 3.1 Sensitivity analysis

An important background step for experiment design is sensitivity analysis. An analysis of the sensitivity functions can provide insight in (i) which parameters have the largest influence and, hence, have to be estimated preferably and (ii) which experimental conditions enclose the most information for the accurate estimation of specific parameters.

#### 3.2 Design of Experiments

The Design of Experiments provides new inputs for the Parameter Estimation. As Design of Experiments involves an optimization problem, first the criterion has to be selected among the available ones (Section 2.2). As a next step the type of input signal as well as the parameters characterizing the signal have to be selected. In the current work the E-criterion:  $\max[\lambda_{\min}(\mathbf{F})]$  is selected with input structures according to the above three strategies.

#### 3.3 Parameter Estimation

Given a designed input, an experiment can be performed and a parameter estimate can be obtained.

Parameter Estimation involves the selection of parameter values such that the model predictions  $\mathbf{y}(\mathbf{p}, t_i)$  fit the measurements  $\mathbf{y}_{exp}(t_i)$  as accurately as possible despite the presence of measurement errors. The most common assumption about the probability distribution of the measurement errors is that they are additive, independent and identically distributed according to a Gaussian distribution. These assumptions typically lead to a *sum of squares* objective (Walter and Pronzato, 1997):

$$J(\mathbf{p}) = \sum_{i=1}^{n_t} \left( \mathbf{y}(\mathbf{p}, t_i) - \mathbf{y}_{exp}(t_i) \right)^T \mathbf{Q} \left( \mathbf{y}(\mathbf{p}, t_i) - \mathbf{y}_{exp}(t_i) \right) \quad (8)$$

with  $n_p$  the number of parameters and  $n_t$  the number of measurements. The *weighting matrix*  $\mathbf{Q}$  is typically selected as the inverse of the measurement error variance-covariance matrix.

### 3.4 Confidence Intervals Computation

The quality of a parameter estimate, or vice versa, its uncertainty, is quantified by its variance or standard deviation, which is a measure for the spread of the parameter distribution. Parameter variances can be extracted from the parameter variance-covariance matrix  $\mathbf{P}$ :

$$\mathbf{P} = E[(\mathbf{p}^* - \hat{\mathbf{p}})(\mathbf{p}^* - \hat{\mathbf{p}})^T] \quad (9)$$

with  $\mathbf{p}^*$  and  $\hat{\mathbf{p}}$  the true and estimated parameter vector, respectively. The variance-covariance matrix  $\mathbf{P}$  can be approximated by the inverse of the *Fisher information matrix* (for more details see Walter and Pronzato (1997)). The variances  $s_i$  on the main diagonal of  $\mathbf{P}$  can be used to determine the  $(1-\alpha)100\%$  confidence interval for each parameter estimate

$$\left[ \hat{p}_i - t_{(1-\frac{\alpha}{2}, n_t - n_p)} \sqrt{s_i^2}, \hat{p}_i + t_{(1-\frac{\alpha}{2}, n_t - n_p)} \sqrt{s_i^2} \right] \quad (10)$$

with  $t$  the Student-t value for  $n_t - n_p$  degrees of freedom,  $(1-\alpha)$  the confidence level,  $n_t$  the number of experimental data points and  $n_p$  the number of estimated parameters.

### 3.5 Implementation

The airpath model is provided as a Simulink file that can be called from Matlab. Due to model specifications the sensitivity functions have to be approximated using a finite difference approach in the current work.

For the PE the `lsqnonlin` Matlab function is used. It is a gradient based method but the gradients are computed using first-order finite difference perturbations. For the Confidence Intervals the Student-t value from Matlab is used, and the variance-covariance matrix is calculated with outputs of the PE procedure. Finally for the OED the `fmincon` Matlab function is used for the optimization whereas the sensitivities are calculated with finite differences as well.

## 4 Results

In this section the results for the airpath model will be presented. First, it has been seen from the sensitivity analysis that all parameters have an influence on at least one of the outputs. Hence, this strengthens the believe that all parameters can be estimated (when appropriate inputs are applied).

However, in view of conciseness, plots of the sensitivities have been omitted. Second, inputs have been optimized based on each of the three strategies. Every strategy was applied on four different operating points (O.P.) as defined in Table 1. Also a comparison to white noise has been made. Third, the designed inputs have been applied to the testbench and parameters have been estimated. Finally, a validation of the calibrated model has been performed.

O.P.	$X_{egr}$ [%]	$X_{vgt}$ [%]	$n[rpm]$	$W_f$ [mg/cyc]
1.	40	85	1500	15
2.	40	75	2000	15
3.	55	80	1800	7
4.	20	70	2200	25

Table 1. Operating points.

### 4.1 Optimization Strategy Comparison

During the entire procedure the sampling time is 0.01 s and the duration of the cycle is 20 s. The frequency resolution is 0.05 Hz. Due to the practical implementation issues the upper bound on the frequency bands are constrained to  $X_{egr} \leq 10$  Hz,  $X_{vgt} \leq 5$  Hz,  $n \leq 2$  Hz and  $W_f \leq 4$  Hz.

The random phases are chosen to be random values in the interval  $[0, 2\pi)$ . The given measurement variance for MAP is  $\sigma_{MAP}^2=100^2$  and for MAF is  $\sigma_{MAF}^2=10^2$ .



**S1 Strategy.** In a first optimization approach only the RMS value of the four multisines added to the operating points is optimized. This means that both the frequency band of the four signals as well as the amplitudes of the different frequency contributions remained the same in the entire frequency range.

The resulting minimal eigenvalues are summarized in Table 2. Operating points 2 and 4 are of higher information.

O.P.	$\lambda_{min}$	RMS value			
		$X_{egr}$	$X_{vgt}$	$n$	$W_f$
1.	3.79e03	12.75	4.78	184.42	4.04
2.	2.92e06	12.24	0.03	300.00	2.52
3.	2.46e03	12.16	7.88	259.66	1.99
4.	1.99e07	5.10	8.00	194.34	1.50

Table 2. RMS optimization results.

**S2 Strategy.** The effect of optimizing the frequency bands is exploited in the second strategy. The idea is that the system can be excited in a certain band. As a result, more information can be obtained by exciting a specific frequency band.

The RMS value and the amplitudes are kept constant in this approach. The RMS values are tuned at the values obtained in the previous strategy. In Table 3 there is an overview of the obtained frequencies as well as the used RMS values. For operating points 1 and 3 a higher  $\lambda_{min}$  than S1 is obtained whereas for operating point 2 and operating pint 4 the same.

**S3 Strategy.** The last approach includes the optimization of the RMS value with varying amplitudes in different frequency bands. Four different sub-bands for each signal are chosen in order not to increase the number of decision variables too much and by doing so to limit the increase in the computation time.

In total for the current application there are 20 optimization variables, as there are four sub-bands for every operating point and four RMS values. The resulting values can be found in Table 4. Operating points 2 and 4 have been improved whereas for operating points 1 and 3 the amplitude optimization does not improve the result of the frequency optimization.

In S3 strategy the frequency region is divided in 4 subbands, which can either fully selected or not whereas in S2 strategy the frequency is a degree of freedom and can vary and thus improving the result (see Figure 2).

By studying the tables for the different strategies it can be seen that the frequency optimized strategy gives results in average of high quality in a reasonable computation time.

#### 4.2 Comparison to White Noise Inputs

For consolidating the above results a comparison with white noise inputs has been carried out. The white noise has been chosen in order to have the same RMS value as the original signal and it is added to the operating points.

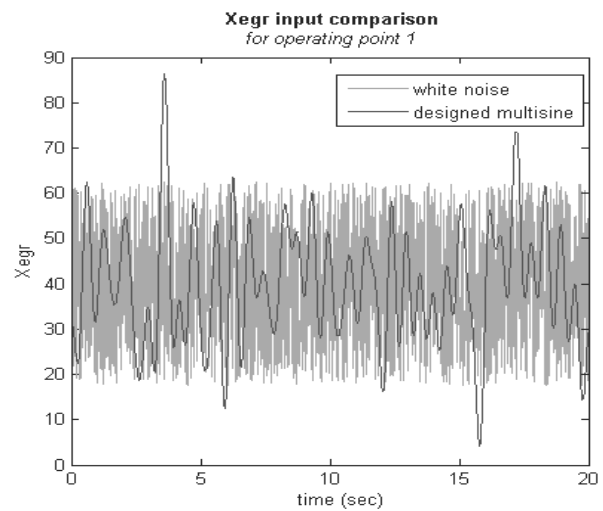


Figure 4. Comparison of  $X_{egr}$  input in multisine and white noise form.

O.P.	$\lambda_{min}$	Frequency band values			
		$X_{egr}$	$X_{vgt}$	$n$	$W_f$
1.	1.39e04	[0.20-1.00]	[0.30-1.06]	[0.05-1.39]	[0.13-1.00]
2.	2.92e06	[0.05-2.00]	[0.05-5.00]	[0.05-2.00]	[0.05-4.00]
3.	7.34e03	[0.43-1.00]	[0.05-1.90]	0.05-1.58]	[0.14-1.00]
4.	1.99e07	[0.06-2.00]	[0.06-5.00]	[0.06-2.00]	[0.06-4.00]

Table 3. Frequency optimization results.

O.P.	$\lambda_{min}$	$X_{egr}$				$X_{vgt}$				$n$				$W_f$			
		RMS	$A_k$			RMS	$A_k$			RMS	$A_k$			RMS	$A_k$		
1.	1.21e04	12.80	0.68/0.98/0.18/0.00			4.98	1.00/ 0.405/ 0.12/ 0.00			190.00	0.85/ 0.00/ 0.48/ 0.12			4.78	0.65/ 0.00/ 0.00/ 0.00		
2.	3.38e06	12.7803	0.00/ 1.00/ 0.00/ 0.95			0.44	0.92/ 0.97/ 0.93/ 0.93			299.99	0.98/ 0.93/ 0.98/ 0.96			2.66	0.95/ 0.96/ 0.92/ 0.97		
3.	2.84e03	12.19	0.97/ 0.99/ 0.98/ 0.80			8.00	1.008/ 0.99/ 0.98/ 0.10			270.00	0.98/ 0.99/ 0.95/ 0.96			1.99	0.97/ 0.99/ 0.96/ 0.96		
4.	2.30e07	5.0967	0.50/ 0.50/ 0.50/ 0.50			8.00	0.50/ 0.50/ 0.50/ 0.50			194.34	0.50/ 0.50/ 0.50/ 0.50			1.50	0.50/ 0.50/ 0.50/ 0.50		

Table 4. Results of RMS optimization with different amplitudes for different frequency bands.

Table 5 summarizes a complete comparison. The minimum eigenvalue for the white noise shown in the table is the mean value of ten different trials in order to ensure an adequate comparison. Figure 4 displays the  $X_{egr}$  input for the S1 and the white noise case. The  $\lambda_{min}$  of the white noise is between 200-600 whereas with the multisines  $10^3 - 10^6$  higher values are achieved.

O.P.	$\lambda_{min}$		
	RMS optimized	frequency optimized	white noise
1.	3.7855e03	1.3893e04	228.05
2.	2.9155e06	1.4303e04	366.23
3.	2.4572e03	7.3435e03	237.59
4.	1.9915e07	1.9915e07	662.76

Table 5. Comparison for optimization of the RMS, frequency and white noise.

### 4.3 Identification

The designed inputs are applied to a real engine. A production 2 liter EU5 common rail Diesel engine mounted on a dynamical engine test-bench was used in the current work. The generated outputs are used for identifying the model's parameters.

The designed inputs were applied on a real system, the measured outputs were used for PE. The resulting parameters were used to simulate outputs. Figure 6 illustrates the engine outputs (measurements) together with the simulated outputs from the designed inputs of the third operating point. The designed input is applied from 5 to 25 seconds. It can be seen that the calibrated model accurately describes the measurements. The resulting parameters together with their confidence bounds can be found in Table 6. The confidence bounds indicate that the estimation is accurate. Which are typically at least the order of magnitude smaller than the parameter values.

### 4.4 Validation

In the above subsection parameters were obtained through identification for OP 4. In the current subsection designed inputs for OP 2 were applied to the real engine, while the simulator used the previously obtained parameters from OP 4 to predict the outputs.

The simulated outputs are following the engine outputs as seen in Figure 6. The SSE value for this estimation is  $SSE = 1.45e+03$ . This validation corroborates the proposed method.

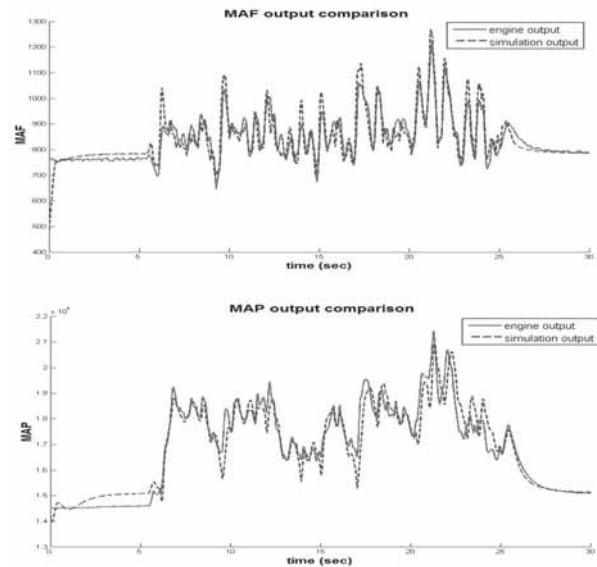


Figure 5. Output comparison.

## 5 Conclusion

In this work the dynamic parameters of an airpath model for a 2 liter EU5 common rail Diesel engine with external exhaust gas recirculation and a variable geometry turbine turbocharger have been accurately estimated. To ensure parameter accuracy, optimal random multisine inputs have been designed based on Optimal Experiment Design techniques.

Parameter Estimation with real engine measurements for operating point 4		
parameter	found value	95% conf.bound
$V_i$	0.0183	$\pm 6.59e-04$
$V_x$	0.0100	$\pm 7.78e-04$
$\eta_n$	0.7098	$\pm 1.48e-03$
$\tau_{egr}$	0.2161	$\pm 9.28e-03$
$\tau_{vgt}$	0.5620	$\pm 9.56e-03$
SSE	1.222e+03	
MSE	0.4078	

Table 6. Test-bench measurements results.

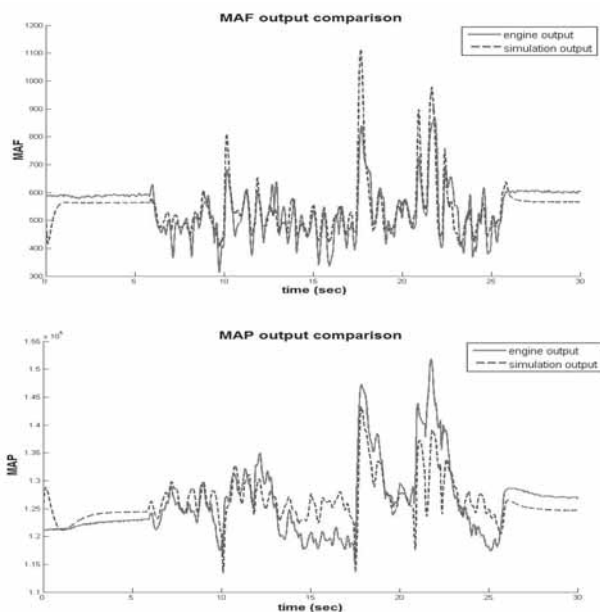


Figure 6. Output comparison validation.

Three input optimization profiles have been presented. The use of multisines allows exclusive excitation of a specific frequency band of interest. Moreover, fast dynamics are included without increasing the number of discretization variables. The results have provided an acceptable prediction both in identification as in validation.

#### ACKNOWLEDGEMENTS

This research is supported by projects OT/10/035, PFV/10/002 (Center-of-Excellence Optimization in Engineering) of the Research Council of the KULeuven, Project KP/09/005 (SCORES4CHEM) of the Industrial Research Council of the

KULeuven, the Belgian Program on Interuniversity Poles of Attraction, initiated by the Belgian Federal Science Policy Office and the industry project ACCM. D.

Telen has a Ph.D grant of the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT- Vlaanderen). J. Van Impe holds the chair Safety Engineering sponsored by the Belgian chemistry and life sciences federation essenscia. The scientific responsibility is assumed by its authors.

#### REFERENCES

- [1] Alberer, D. (2009). *Fast Oxygen Based Transient Diesel Engine Control*. Ph.D. thesis, J.K. Universität Linz.
- [2] Balsa-Canto, E., Alonso, A., and Banga, J. (2010). An iterative identification procedure for dynamic modeling of biochemical networks. *BMC Systems Biology*, 4.
- [3] Biegler, L. (2007). An overview of simultaneous strategies for dynamic optimization. *Chemical Engineering and Processing: Process Intensification*, 46, 1043–1053.
- [4] Diehl, M., Bock, H., Schlöder, J., Findeisen, R., Nagy, Z., and Allgöwer, F. (2002). Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations. *Journal of Process Control*, 12, 577–585.
- [5] Ferreau, H.J., Ortner, P., Langthaler, P., del Re, L., and Diehl, M. (2007). Predictive control of a real-world diesel engine using an extended online active set strategy. *Annual Reviews in Control*, 31(2), 293–301.
- [6] Franceschini, G. and Macchietto, S. (2008). Model-based design of experiments for parameter precision: State of the art. *Chemical Engineering Science*, 63, 4846–4872.
- [7] Ljung, L. (1999). *System Identification: Theory for the User*. Prentice Hall.
- [8] Pintelon, R. and Schoukens, J. (2001). *System Identification, A Frequency Domain Approach*. IEEE Press.
- [9] Puchner, S., Winkler-Ebner, B., Alberer, D., and del Re, L. (2009). Optimization based mean value model of turbocharged diesel engines. In *MATHMOD 2009 - 6th Vienna Int. Conference on Mathematical Modelling*.
- [10] Stewart, G., Borrelli, F., Pekar, J., Germann, D., Pachner, D., and Kihás, D. (2011). *Automotive Model Predictive Control: Models, Methods and Applications*, chapter Towards a Systematic Design for Turbocharged Engine Control, 211. Springer.
- [11] Walter, E. and Pronzato, L. (1997). *Identification of Parametric Models from Experimental Data*. Springer, Paris.

Submitted MATHMOD 2012: November 2011

Accepted MATHMOD 2012: January 2012

Submitted SNE Revised: October 2012

Accepted: December 5, 2012

# Predictions of Carbon Fixation during a Bloom of *Emiliana huxleyi* as a Function of the Regulating Inorganic Species

Olivier Bernard<sup>1\*</sup>, Antoine Sciandra<sup>2</sup>, Sophie Rabouille<sup>2</sup>

<sup>1</sup>COMORE-INRIA, BP93, 06902 Sophia-Antipolis Cedex, France; \*[olivier.bernard@inria.fr](mailto:olivier.bernard@inria.fr)

<sup>2</sup>LOV, UMR 7093, Station Zoologique, B.P. 28 06234, Villefranche-sur-Mer, France

**Abstract.** Large scale precipitation of calcium carbonate in the oceans by coccolithophorids is a phenomenon that plays an important role in carbon sequestration. However, there is a controversy on the effect of an increase in atmospheric CO<sub>2</sub> concentration on both calcification and photosynthesis of coccolithophorids. Indeed recent experiments, performed in conditions of nitrogen limitation, revealed that the associated fluxes may be slowed down, while other authors claim the reverse response. We have designed models to account for various scenarii of calcification and photosynthesis regulation in chemostat cultures of *Emiliana huxleyi*, based on different hypotheses of regulation mechanism. These models, which are kept at a general and generic level, consider that either carbon dioxide, bicarbonate, carbonate or pH is the regulating factor.

These models are calibrated to predict the same carbon fluxes in nowadays pCO<sub>2</sub>, but they turn out to respond differently to an increase of CO<sub>2</sub> concentration. Thus, we simulated a bloom of *Emiliana huxleyi* using the four considered regulation scenarii. For high biomass concentration, the coccolithophorids can significantly affect the inorganic carbon and the pH in their environment, thus leading to a feedback in their growth rate which is, depending on the model, positive or negative. It results that the prediction of the carbon fixed during the bloom varies by a factor 2, depending on the assumed regulating mechanism hypothesized for growth and calcification.

## Introduction

Phytoplankton uses light energy to build up organic cell components from inorganic carbon, and thus participates in the so-called 'biologic pump' that traps CO<sub>2</sub> from the atmosphere. In the world oceans, the activity of phytoplankton accounts for about 40 % of the total, primary production on Earth.

As pCO<sub>2</sub> levels in the atmosphere rise, phytoplankton growth might be positively stimulated by an increased availability of dissolved CO<sub>2</sub> in the upper oceans. However, a trade-off appears between CO<sub>2</sub> being more available for growth, and a lowered pH due to the chemical equilibrium of the carbonate system and the consequent ocean acidification.

Coccolithophorids are particularly abundant in the oceans and thus play an important role in CO<sub>2</sub> trapping [6]. These organisms are remarkable by the presence of solid, calcite structures called coccoliths that surround their cell. Coccolithophorids hence account for up to a third of the total, marine CaCO<sub>3</sub> production. Such structures are relatively sensitive to pH and tend to dissolve when the water becomes too acidic. It is expected that increases in pCO<sub>2</sub> will have direct consequences on the ability of these organisms to maintain their growth rate. As a corollary, acidification of the oceans due to atmospheric pCO<sub>2</sub> increases could jeopardize their role as a CO<sub>2</sub> pump.

Hence, how Coccolithophorids may respond to shifts in global pCO<sub>2</sub> is a critical question to be answered. However, if the photosynthesis mechanisms are well known, the effects of pCO<sub>2</sub> changes, whether on photosynthesis or on calcification, are still subject to intense debates. In batch experiments, contradictory observations have been made, where increases in pCO<sub>2</sub> either led to a decrease [8] or an increase [7] in calcification in *Emiliana huxleyi*, while photosynthesis was enhanced. Continuous cultures experiments in chemostats supported the hypothesis that both photosynthesis and calcification decrease [9].

In this paper, we investigate the relations between photosynthesis and calcification. We present a set of models, extended from [1], that integrate both phytoplankton growth and the carbonates system dynamics in the water.

They were specifically designed to test several possible couplings and regulation mechanisms, assuming that calcification is regulated by one of the chemical species among  $CO_2$ ,  $HCO_3^-$  and  $CO_3^{2-}$ . The model, based on the representation of a cell quota, is a Droop-like model [3, 2, 4] that we kept as general and generic as possible. Then, we add a fourth model where the calcite dissolution state acts as a regulating factor.

To complete these biological models, a simplified representation of the carbonate system is proposed with three equations. Hence, knowing the concentration of dissolved inorganic carbon (DIC), the concentration in  $Ca^{2+}$  and considering the hypothesis of a constant concentration of the other ions in the water, the seawater model can predict the pH value and concentrations of  $CO_2$ ,  $HCO_3^-$  and  $CO_3^{2-}$ . This leads to four possible simplified models that can each represent a bloom of E.hux. These models bring two noteworthy results. We show that the predicted biomass can vary two-fold depending on the model, and that  $pCO_2$  has little influence on the bloom, due to the slow transfer of inorganic carbon at the atmosphere – seawater interface.

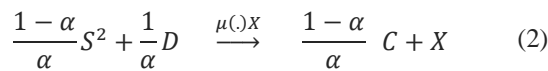
## 1 Modelling Growth and Calcification

### 1.1 Biological Aspects

Here we present the mass flows in the model corresponding to nitrogen and carbon uptake. Uptake of inorganic nitrogen (nitrate, denoted  $S_1$ ) into the phytoplanktonic biomass (whose particulate nitrogen concentration is denoted  $N$ ), can be represented by the following mass flow, where  $\rho(\cdot)$  is the nitrate absorption rate:



The flux of inorganic carbon into organic biomass  $X$  and coccoliths  $C$  is associated to a flux of calcium ( $Ca^{2+}$ , denoted  $S_2$ ):



There  $\mu(\cdot)$  is the photosynthesis rate.

The next question is the modelling of both the nitrate absorption rate  $\rho(\cdot)$  and the photosynthesis rate  $\mu(\cdot)$ .

Generally, the nitrate uptake rate is assumed to depend on external nitrate concentration  $NO_3$ , following a Michaelis-Menten type equation [5].

The expression of the rate of inorganic carbon acquisition is trickier, as shown by [3, 4], it must depend on the internal nitrogen quota  $Q$ . However, coccolithophorids photosynthesis and calcification are also sensitive to the DIC concentration, and there is a consensus to admit that  $CO_2$  is the substrate for photosynthesis while  $HCO_3^-$  is the substrate of calcification. Therefore the regulation of growth and calcification can theoretically be triggered by  $CO_2$  or  $HCO_3^-$ . We also examine the possibility that  $CO_3^{2-}$  is involved in the regulation process of inorganic carbon acquisition [1]. Finally, we also propose in this paper to consider the availability of calcium as a possible regulating factor of photosynthesis and calcification. In this last hypothesis, we examine the possibility that  $\mu(\cdot)$  is regulated by  $\Omega$ , the saturation state of calcite ( $CaCO_3$ ); there solubility constant yields  $K_{sp}=5.1510^{-7} \text{mol}^2.L^{-2}$ :

$$\Omega = \frac{[Ca^{2+}][CO_3^{2-}]}{K_{sp}} \quad (3)$$

As a consequence, in the sequel we examine four possible models that only differ by the regulation mechanisms of inorganic carbon acquisition.

- $CO_2$  is the regulating species, and thus  $\mu(Q, CO_2)$  is an increasing function of both  $Q$  and  $CO_2$ .
- $HCO_3^-$  is the regulating species, and thus  $\mu(Q, HCO_3^-)$  is an increasing function of both  $Q$  and  $HCO_3^-$ .
- $CO_3^{2-}$  is the regulating species, and thus  $\mu(Q, CO_3^{2-})$  is an increasing function of both  $Q$  and  $CO_3^{2-}$ .
- $\Omega$  is the regulating species, and thus  $\mu(Q, \Omega)$  is an increasing function of both  $Q$  and  $\Omega$ .

To keep a general denomination, we denote  $\mu_p(Q, D_p)$  the growth rate, where, depending on the model  $M_p$ ,  $D_p$  has to be chosen  $CO_2$ ,  $HCO_3^-$ ,  $CO_3^{2-}$  and  $\Omega$ .

For simulation purposes, we represent the  $NO_3$  uptake rate [5],  $\rho(S_1) = \rho_m S_1 / (S_1 + k_N)$ , where  $\rho_m$  and  $k_N$  are the maximum uptake rate and the half-saturation constant, respectively. Based on the Droop model [3, 4], the net growth rate may be written as:

$$\mu(Q, D_p) = \bar{\mu} \left( 1 - \frac{k_Q}{Q} \right) \frac{D_p}{D_p + k_{D_p}} - R \quad (4)$$

where  $k_Q$ ,  $\mu$  and  $k_{D_p}$  are respectively the subsistence internal quota, the maximum hypothetical growth rate and the half-saturation constant for the chosen regulating species.  $R$  is the respiration rate (supposed to be constant).

### 1.1 Seawater Modelling

In order to compute  $CO_2$ ,  $HCO_3^-$ ,  $CO_3^{2-}$  and  $\Omega$  from  $D$  and  $S_2$ , classical equations of the seawater carbonate system must be considered [10]. The carbonate alkalinity (CA) represents the electric charges carried in the carbonate system:

The total alkalinity (TA) is defined by (see [10] for more details):

$$CA = [HCO_3^-] + 2[CO_3^{2-}] \quad (5)$$

The total alkalinity (CA) is defined by (see [10] for more details):

$$TA = CA + [B(OH)_4^-] + [OH^-] - [H^+] \quad (6)$$

We denote  $\lambda = TA - 2[Ca^{2+}] = TA - 2S_2$ . In a first approximation, the ions that most contribute to  $\lambda$  depend on the salinity and remain constant.

Following the previous considerations, carbonate alkalinity thus only depends on calcium:  $CA = \lambda - \lambda_0 + 2S_2$  (where, in a first approximation,  $\lambda_0 = [B(OH)_4^-] + [OH^-] - [H^+]$  remains constant compared to CA). In order to compute the  $[HCO_3^-]$  and  $[CO_3^{2-}]$  concentrations, we use the dissociation constants of the carbon dioxide ( $K_1$ ) and bicarbonate ( $K_2$ ) (the proton concentration,  $[H^+]$ , will be denoted  $h$ ):

$$K_1 = \frac{h[HCO_3^-]}{[CO_2]}, \quad K_2 = \frac{h[CO_3^{2-}]}{[HCO_3^-]} \quad (7)$$

The total dissolved inorganic carbon (D) is defined as:

$$D = [HCO_3^-] + [CO_3^{2-}] + [CO_2] \quad (8)$$

Note that, in the considered pH range, we have  $[HCO_3^-] \gg [CO_3^{2-}] \gg [CO_2]$  (see for example [10]). It follows that bicarbonate is the main carbon species in the bicarbonate system:

$$[HCO_3^-] \cong D \quad (9)$$

We deduced from equations (5) and (8), in the considered pH range:

$$[CO_3^{2-}] \cong CA - D \quad (10)$$

With this approximation, we can now compute the following ratio:  $r = \frac{D}{CA}$ , using equations (5), (8) and (7), we get:

$$r = \frac{h + K_2 + h^2/K_1}{h + 2K_2} \quad (11)$$

It follows that  $h$  can be computed as a function of  $r$ :

$$h = u(r) = \left( -1 + r + \sqrt{(1 - 2r) \left( 1 - \frac{4K_2}{K_1} \right) + r^2} \right) \frac{K_1}{2} \quad (12)$$

Now using equations (7) and (5) we can compute the exact  $CO_2$  concentration:

$$[CO_2] = \frac{CA}{K_1} \frac{h^2}{h + 2K_2} = CA v(r) = \psi(S_2, D) \quad (13)$$

This simplified seawater modelling allowed a mathematical analysis of coccolithophorids models [1]. However, in the simulation, we used a more accurate model that does not make any approximation. The used Matlab code is a supplement to [10].

## 2 Modelling of a *E. Huxleyi* Bloom in a Mixed Layer

In summer, increasing temperatures lead to a density gradient that stabilizes the water column, which then stratifies. The surface layer remains mixed over a generally shallow depth (in the order of 20m) and to keep the model as simple as possible, we assume a homogeneous distribution. We simulated the growth of coccolithophorids in this mixed layer, as represented in Figure 1.  $CO_2$  concentration in the water equilibrates with that in the atmosphere, following the difference in concentration between the two compartments and according to the diffusion coefficient  $K_L a$ .

Diffusion at the ocean surface is generated by wind stress, and so much lower  $K_L a$  values must be considered here compared to *e.g.* bioreactors. That is, the low value ( $0.06 \text{ day}^{-1}$ ) used in the model is representative of the natural environment.

As a corollary, it is expected that high biomasses may draw down the DIC pool faster than it is renewed. In the water,  $CO_2$  equilibrates with  $HCO_3^-$  and  $CO_3^{2-}$ . The  $CO_2$  pool in the water is also affected by the coccolithophorids activity, being fuelled by respiration and consumed through the growth process (see (2)).

The model simulates a nitrate uptake limited by the availability of  $NO_3^-$ , as illustrated by (1), while growth and coccoliths formation depend on the availability of both  $Ca^{2+}$  and  $CO_3^{2-}$  (see (2)).  $NO_3^-$  and  $Ca^{2+}$  are provided by upwelling of deeper waters underlying the mixed layer (with an exchange rate  $K_d$ ).

The water acidity affects the coccoliths persistence; we accounted for a possible dissolution of coccoliths, whose rate is dependent upon pH and represented by  $\frac{K_{diss}}{\Omega} C$ .

Settlement of calcite (detached coccoliths) is represented through  $CaCO_3$  sinking below the mixed layer.

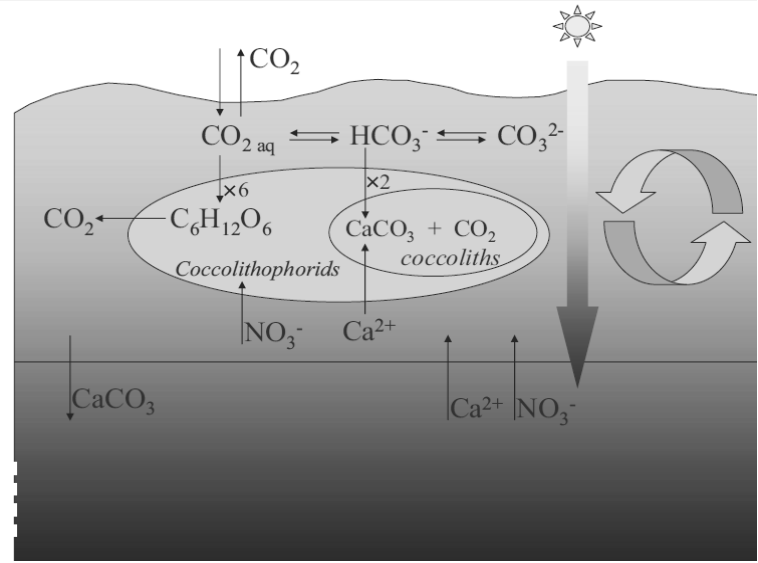


Figure 1. Schematic diagram of the well mixed upper ocean represented by the model.

Model equations can then be directly deduced from the mass flows (1) and (2).  $D_p$  is the regulating factor (among  $CO_2$ ,  $HCO_3^-$ ,  $CO_3^{2-}$  and  $\Omega$ ) assumed to regulate both photosynthesis and calcification. The system of equations reads:

$$\dot{S}_1 = K_d(S_{10} - S_1) - \rho(S_1)X \quad (14)$$

$$\dot{Q} = \rho(S_1) - \mu(Q, D_p) Q \quad (15)$$

$$\dot{X} = -K_d X + \mu(Q, D_p) X - R X - K_{sed} X \quad (16)$$

$$\dot{C} = -K_d C + \frac{1 - \alpha}{\alpha} \mu(Q, D_p) X - K_{sed} C - \frac{K_{diss}}{\Omega} C \quad (17)$$

$$\begin{aligned} \dot{D} = & K_d(D_0 - D) - \frac{1}{\alpha} \mu(Q, D_p) X - R X - \\ & - K_L \alpha (\psi(S_2, D) - K_H p CO_2) + \frac{K_{diss}}{\Omega} C \end{aligned} \quad (18)$$

$$\dot{S}_2 = K_d(S_{20} - S_2) - \frac{1 - \alpha}{\alpha} \mu(S_1) X \quad (19)$$

There the exchange rate at the thermocline level is  $K_d$ , the sedimentation rate is  $K_{sed}$ , and the coccoliths dissolution rate is  $\frac{K_{diss}}{\Omega}$ .

The specific rate of carbon fixation is described as an increasing function of  $Q$  and  $D_p$ , which allows a generic analysis of the model [1]. Depending on the choice for  $D_p$ , four different models are obtained, based on three different hypotheses on the mechanisms driving both photosynthesis and calcification.

Parameters	Values	Units
$S_{10}$	50	$\mu mol N.L^{-1}$
$S_{20}$	10.4	$mmol Ca.L$
$D_0$	1.77	$mmol C.L^{-1}$
$K_L a$	0.06	$d^{-1}$
$\rho_m$	100.19	$\mu mol N.mmol C^{-1}.d^{-1}$
$k_Q$	32.29	$\mu mol N.mmol C^{-1}$
$k_{S_1}$	0.038	$\mu mol N.L^{-1}$
$K_1$	$1.392 \cdot 10^{-6}$	$mmol.L^{-1}$
$K_2$	$1.189 \cdot 10^{-9}$	$mmol.L^{-1}$
$K_H$	36.7	$mmol CO_2.L^{-1}.\mu atm$
$\alpha$	0.53	—
$\lambda$	$-17.31^3$	$mmol.L^{-1}$
$\lambda_0$	$0.086^3$	$mmol.L^{-1}$
$K_{diss}$	0.15	$d^{-1}$
$K_d$	0.8	$d^{-1}$
$K_{sed}$	0.15	$d^{-1}$
$R$	0.01	$d^{-1}$

Table 1. Values of the model parameters.



Parameters	$CO_3^{2-}$	$HCO_3^-$	$CO_2$	$\Omega$	Units
$k_{D_p}$	0.076	1.65	0.01	1.53*	$\mu mol C.L^{-1}$
$\bar{\mu}$	2.83	3.76	3.24	2.88	$d^{-1}$

**Table 2.** Kinetics parameters depending on the chosen model. (\* unitless for  $k_{D_p}$ ).

The models have been calibrated in order to predict the same carbon fluxes in nowadays  $pCO_2$ , on the basis on available experimental results [1]. Parameter values are presented in Table 1 and Table 2.

The model analysis proposed in [1] demonstrates that  $M_p$  models where  $D_p$  is either  $CO_2$  or  $HCO_3^-$  support the results of [7], while models where  $CO_3^{2-}$  or  $\Omega$  is the regulating factor support the results obtained by [9]. Last, none of these models allowed a qualitative prediction of the experimental results reported by [8]. Different model hypotheses were then required to reproduce these observations: photosynthesis had to be regulated by either  $CO_2$  or  $HCO_3^-$  while calcification was driven by  $CO_3^{2-}$  or  $\Omega$  [1].

### 3 Model Simulation

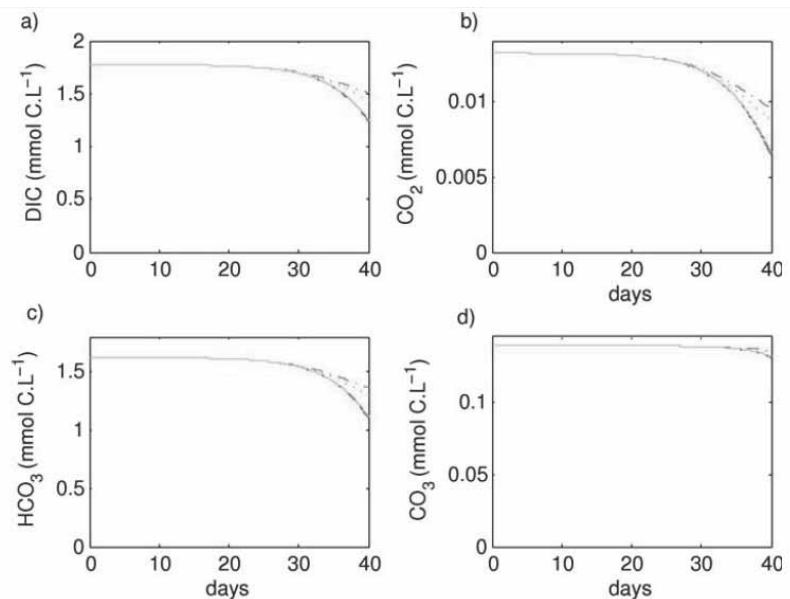
We used each of these models to simulate a large development (bloom) of *Emiliana huxleyi*. Phytoplankton cells are assumed to grow in a homogeneous layer, where aqueous  $CO_2$  is in equilibrium with the atmosphere. The considered, realistic  $K_{La}$  being rather low, the time necessary to supply inorganic carbon to the cells can be long. This can explain the significantly different behaviour between the 4 scenarii (Figure 2).

Indeed, it turns out that, for high biomass concentrations, the coccolithophorids can significantly draw down the inorganic carbon and thus affect the pH in their environment. Depending on the model, the simulated mechanisms induce a positive (in the models with  $CO_3^{2-}$  or  $\Omega$  as regulating factor) or negative (in models with  $CO_2$  or  $HCO_3^-$ ) feedback on the growth rate.

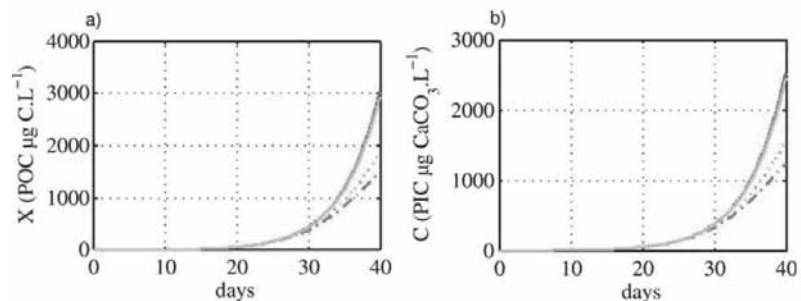
It results that the prediction of carbon fixed during the bloom formation can vary by a factor up to 2, depending on the assumed regulating mechanism hypothesized for growth and calcification (Figure 3).

The simulations with  $\Omega$  as regulating factor make little difference to that with  $CO_3^{2-}$ . Such result can be explained by the fact that changes in  $Ca^{2+}$  concentrations being small,  $\Omega$  fluctuations are similar to that of  $CO_3^{2-}$ .

When introducing a coccoliths dissolution term, model results remain close to that obtained without dissolution rate. Hence, the rate of coccoliths dissolution stayed low.



**Figure 2.** Depending on the considered choice of  $D_p$  ( $CO_2$ : -.-,  $HCO_3^-$ : .....,  $CO_3^{2-}$ : —,  $\Omega$ : -.-.-), evolution of the various compartment of inorganic carbon. The model where  $HCO_3^-$  is the regulating factor shows the strongest impact on the inorganic carbon.



**Figure 3.** Amount of inorganic carbon which is fixed through photosynthesis (a) and calcification (b), depending on the considered ( $CO_2$ : -.-,  $HCO_3^-$ : .....,  $CO_3^{2-}$ : —,  $\Omega$ : -.-.-). The models where  $D_p = CO_3^{2-}$  or  $D_p = \Omega$  predict much higher carbon fluxes.

This can be explained by the remarkable stability of  $CO_3^{2-}$  whose concentration variation did not exceed 10%. Indeed the decrease of  $CO_3^{2-}$  due to exhaustion of total inorganic carbon is compensated by the pH increase that favors the form  $CO_3^{2-}$  to the detriment of  $HCO_3^-$ .

Last, investigating the influence of different surface  $pCO_2$  revealed very little impact on growth. An increase from 380 ppm to 600 ppm only modified the total production by about 2%. At the air/sea interface, low  $K_La$  values limit the increase in dissolved  $CO_2$  concentrations and, as a corollary, short time scales (month) changes in  $pCO_2$  in the water do not reflect that in the atmosphere. Consequently, model results suggest that biomass production remains relatively insensitive to changes in atmospheric  $pCO_2$ .

## 4 Conclusion

This study stresses how a correct identification of the chemical species that drive(s) calcification and photosynthesis processes is critical to accurately predict a bloom of coccolithophorids and the consequent amount of carbon withdrawn from the atmosphere and trapped into the deep ocean. The model results reveal a striking difference in the predicted biomass increase when the saturation state  $\Omega$  (or equivalently  $CO_3^{2-}$ ) is the regulating factor.

In the configuration of a low air/sea exchange, model results suggest that increased  $pCO_2$  in the air show very little impact on growth. Due to the exhaustion of the DIC pool by the high biomasses formed during the bloom and low transfer coefficient, changes in surface  $pCO_2$  hardly affect the bloom intensity. Such paradoxical transient behaviour only apply to off shore marine systems. Coastal, shallow ecosystems may present higher diffusion rates and model results then suggest a higher impact of surface  $pCO_2$  on growth: under conditions of higher  $K_La$  values, the  $CO_2$  resupply to the water participates in enhancing bloom formations for models regulated by  $CO_2$  or  $HCO_3^-$  and shows a positive effect on growth, while the opposite behaviour is observed for models regulated by  $CO_3^{2-}$  or  $\Omega$ .

## Acknowledgements.

The authors benefited from the support of the BOOM project (Biodiversity of Open Ocean Microcalcifiers) funded by the French National Research Agency (ANR), and of the European FP7 Integrated Project EPOCA (European Project on Ocean Acidification).

## References

- [1] Bernard, O., Sciandra, A., and Madani, S. (2008). Multimodel analysis of the response of the coccolithophore *Emiliania huxleyi* to an elevation of  $pCO_2$  under nitrate limitation. *Ecol. Model.*, 211, 324–338.
- [2] Burmaster, D. (1979). The unsteady continuous culture of phosphate-limited monochrysis lutheri droop : Experimental and theoretical analysis. *J. of Experimental Marine Biology and Ecology*, 39 (2), 167–186.
- [3] Droop, M. R. (1968). Vitamin B12 and marine ecology. IV. the kinetics of uptake growth and inhibition in *Monochrysis lutheri*. *J. Mar. Biol. Assoc.*, 48(3), 689–733.
- [4] Droop, M. R. (1983). 25 years of algal growth kinetics, a personal view. *Botanica marina*, 16, 99–112.
- [5] Dugdale, R. C. (1967). Nutrient limitation in the sea: dynamics, identification and significance. *Limnol. Oceanogr.*, 12, 685–695.
- [6] Frankignoulle, M., Canon, C., and Gattuso, J. P. (1994). Marine calcification as a source of carbon dioxide: positive feedback of increasing atmospheric  $CO_2$ . *Limnol Oceanogr.*, 39, 458–462.
- [7] Iglesias-Rodriguez, M. D., Halloran, P. R., Rickaby, R. E. M., Hall, I. R., Colmenero-Hidalgo, E., Gittins, J. R., Green, D. R. H., Tyrrell, T., Gibbs, S.J., von Dassow, P., Rehm, E., Armbrust, E. V., Boessenkool, K. P. (2008). Phytoplankton calcification in a high- $CO_2$  world. *Science*, 2008.
- [8] Riebesell, U., Zondervan, I., Rost, B., Tortell, P.D., Zeebe, R. E., and Morel, F. M. M. (2000). Reduced calcification of marine plankton in response to increased atmospheric  $CO_2$ . *Nature*, 407, 364–367.
- [9] Sciandra, A., Harlay, J., Lefèvre, D., Lemée, R., Rimmelin, P., Denis, M., and Gattuso, J.-P. (2003). Response of coccolithophorid *emiliania huxleyi* to elevated partial pressure of  $CO_2$  under nitrogen limitation. *Mar. Ecol. Prog. Ser.*, 261, 111–122.
- [10] Zeebe, R. E., and Wolf-Gladrow, D. (2003). *CO2 in seawater: equilibrium, kinetics, isotopes*. Elsevier.

Submitted MATHMOD 2009: November 2008

Accepted MATHMOD 2009: January 2009

Submitted SNE: January 2010

Accepted: February 5, 2010

# Dynamic Modeling of a Methanization Plant

Andre Bader<sup>1\*</sup>, Sindy Bauersfeld<sup>1</sup>, Christian Brunhuber<sup>1</sup>, Robert Pardemann<sup>2</sup>, Bernd Meyer<sup>1</sup>

<sup>1</sup> TU Bergakademie Freiberg, Department of Energy Process Engineering and Chemical Engineering, Fuchsmühlenweg 9, Haus 1, 09596 Freiberg, Germany, \*Andre.Bader@iec.tu-freiberg.de

<sup>2</sup> Siemens AG, Energy Solutions Freyeslebenstraße 1, 91058 Erlangen, Germany

**Abstract.** The chemical and physical modeling and transient simulation of a methanization plant with chemical reactors is useful for dimensioning, optimization, operation and analyzing of critical processes. The paper introduces the results of the development of a dynamic model for a commercial methanization process. The discussed models base on the free Modelica language and Dymola as user interface. The methanization plant consists of 3 adiabatic fixed bed reactors and steam generation units for heat recovery. Calculation results are shown for the dynamic behavior of the methanization plant at a load change.

**Keywords:** chemical reactor, thermodynamic equilibrium, dynamic simulation, part load, heat recovery, CH<sub>4</sub>.

## Introduction

It is well known that the oil and natural gas reserves are limited. The production of fuels is based almost completely on oil and natural gas. Hence there is the wish to extend availability of a secure energy supply. One solution is the conversion of coal or biomass to synthetic or substitute natural gas (SNG) by gasification and methanization to substitute the rising natural gas need. The SNG can be used in electricity generation in combined-cycle power plant stations. Both coal and biomass are suitable as the reserves of coal will subsist more than 150 years and biomass is a renewable energy source. The SNG synthesis consists of the three main steps gasification, gas cleaning & conversion and the methanization.

The Gasifier was modeled already by Fahlke in [1] and the cleaning and conversion step of the synthesized gas (Syngas) was modeled by Heil in [2]. This paper focuses on the methanization step where the Syngas composition is changed to the SNG composition in order to substitute natural gas which is shown in Table 1.

The simulation of a methanization plant is in interest for plant manufactures and operators to optimize the construction and the operation. Further interests are studies of critical processes for understanding of complex physical and chemical processes during part load and plant faults in order to increase the reliability and availability of the plant.

Gas species	Syngas Vol.-% [3]	SNG Vol.-% [4]	Natural gas Vol.-% [5]
CH <sub>4</sub>	0...20	94...98	85...98
H <sub>2</sub>	12...45	0...2	0
CO	6...58	0	0
CO <sub>2</sub>	5...33	0...2	0...2

Table 1. Gas composition of Syngas, SNG and natural gas.

## 1 Process Description

The raw gas for the SNG synthesis contains mainly H<sub>2</sub>, CO, CO<sub>2</sub>, H<sub>2</sub>O, CH<sub>4</sub>, N<sub>2</sub>. This composition of the raw gas depends on the gasification technology [3]. The SNG synthesis is a heterogeneously catalyzed process. During the methanization, the following chemical reactions occur. The hydrogenation of carbon oxides to methane are the so called CO methanization reaction in equation (R-1) and the CO<sub>2</sub> methanization reaction in equation (R-2) [3].

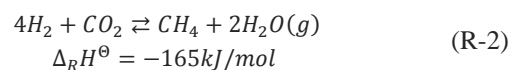
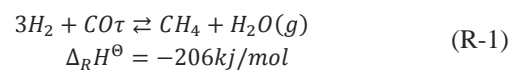
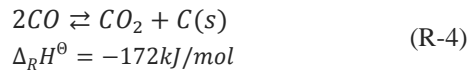
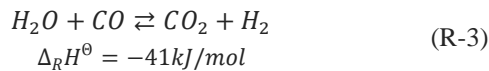


Figure 1. SNG synthesis concept.

Two further independent reactions are important: the homogenous water gas reaction in equation (R-3) and the Boudouard reaction in equation (R-4).



All the reactions of the methanization are exothermic. Hence, the methanization is favoured at low temperatures. Furthermore the methanization is benefited at high pressure, as the reactions (R-1) and (R-2) execute with volume decreasing. The major criterion is to achieve efficient removal of heat for the catalytic methanization reactor system.

The first reason is to minimize catalyst deactivation due to thermal stress. The second reason is to avoid a limitation in the methane yield due to approaching the chemical equilibrium. The Topsøe recycle energy-efficient methanization process (TREMP) from Haldor Topsøe is a suitable reactor concept for the production of SNG [4], which is used for the modeling.

The TREMP consists of three adiabatic fixed bed reactors with gas recycle cooling and interstage gas cooling. Efficient recovery of the reaction heat is essential for the industrial methanization technology. That's why a counter current water cycle stream is used for the interstage cooling. The water cycle is simultaneously used for high pressure steam generation. A gas-recycle back to the inlet of at the first reactor limits the yield and the temperature increase in the reactor [6]. This is necessary to avoid catalysts sintering.

## 2 Model Description

The aim of the modeling is to implement a TREMP methanization plant library with physical based models of components in the Siemens Modelica/Dymola library. The implemented library is based on the *Modelica Fluid Library*, which offers a modeling environment with respect to the implementation of the three balance equations and the interaction through the fluid ports.

The created library enables investigations of a reactor, a heat exchanger, a simple pump, a flash, and a gas and water splitter. The used model components base on the Modelica library, whereas the thermodynamic equilibrium model is a new development.

### Reactor Model

The methanization reactor yield depends on the thermodynamic equilibrium, which will be achieved both in the full and the part load of the methanization plant, Harms in [6].

The thermodynamic equilibrium constant  $K_a$  is used along the law of mass action to determine the molar fractions of the components and the yield of the reactions at the thermodynamic equilibrium.  $K_p$  is the equilibrium constant of the partial pressures,  $p_i$ , of the species, which is connected to the  $K_a$  with the fugacity coefficients,  $K_\alpha$ , in Equation 1.

$$K_p = K_a / K_\alpha \quad (1)$$

The law of mass action for the  $K_p$  is described in equation 2 for a chemical reaction like in equation 3.

$$K_p = \frac{p_C^v \cdot p_D^{v_D}}{p_A^v \cdot p_B^{v_B}} \quad (2)$$

$$v_A A + v_B B = v_C C + v_D D \quad (3)$$

The equilibrium constant of the molar fractions ( $K_x$ ) is necessary to calculate the chemical equilibrium.  $K_x$  can be computed with  $K_p$ , the total pressure,  $p$ , and the sum of stoichiometric coefficients,  $v_i$ , of the chemical reaction, as it is shown in Equation 4.

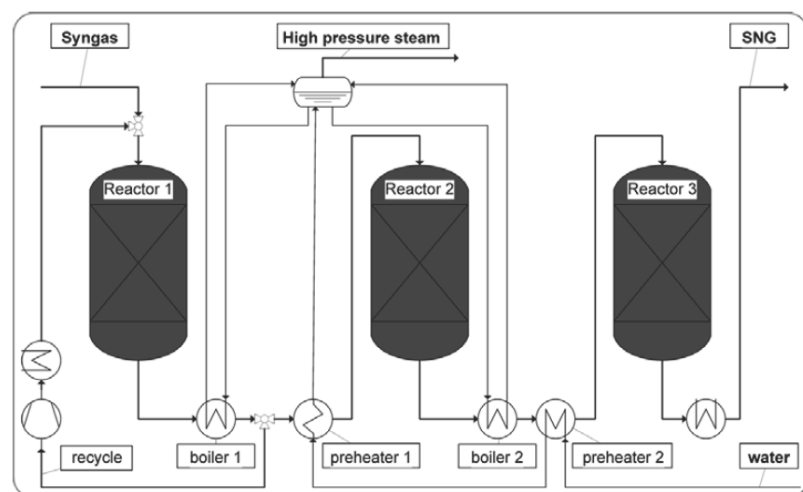


Figure 2. Flow sheet of the Topsøe recycle energy efficient methanization process.

The molar fraction of the species,  $X_i$ , can be determined with equation 5.

$$K_X = K_p \cdot p^{-\sum v_i} \quad (4)$$

$$K_X = \frac{x_C^{v_C} \cdot x_D^{v_D}}{x_A^{v_A} \cdot x_B^{v_B}} \quad (5)$$

It has to be noted that the thermodynamic equilibrium constant depends on the temperature. The temperature dependency can be approximated with Equation 6. The parameters A and B of equation 6 are given in Table 2.

Two simplifications were assumed for the calculation of the thermodynamic equilibrium. The first is the assumption of the ideal gas law for the species. There-with the fugacity coefficients can be neglected in Equation 1,  $K_a = 1$ . The second simplification is the neglect of the Boudouard reaction, Equation (R-4), which will occur if the stoichiometric ratio of Equation 7 is lower than 3. In that case, carbon is produced as a product of reaction (R-4). Carbon leads to catalyst deactivation by deposition on the catalyst surface.

$$\ln(K_A) = A + B/T \quad (6)$$

reaction	A	B
(R-1)	-29.3014	26248.4
(R-2)	-4.3537	4593.2

Table 2. Parameters A and B of equation 6 for reaction (R-1) and (R-3) [7]

$$\frac{H_2 - CO}{CO + CO_2} \geq 3 \quad (7)$$

If the stoichiometric ratio of the reactants of Equation 7 is at least three or more, CO reacts with H<sub>2</sub> completely to CH<sub>4</sub> and H<sub>2</sub>O according to reaction (R-1). The stoichiometric ratio of Equation 7 can be adjusted by converting CO with H<sub>2</sub>O to CO<sub>2</sub> and H<sub>2</sub> as long as the H<sub>2</sub>O content is high enough, see Anderlohr in [8]. Hence the ratio of Equation 7 has to be higher than three in technical processes.

Finally the system of methanization reactions can be reduced to reaction (R-1) and (R-3), as the stoichiometric ratio of Equation 7 for all the gas streams in the TREMP is higher than three, as well.

Therewith the molar fraction of the species can be determined with the equations 8 to 13, in which 0 stands for the initial state and 1 for the equilibrium state. The conversion  $U_1$  and  $U_2$  is the conversion of the reactions (R-1) and (R-3), with their sum being equal to the total conversion of CO, see Equation 14.

$$X_{H_2,1} = \frac{c_{H_2,0} - 3c_{CO,0} \cdot U_1 + c_{CO,0} \cdot U_2}{1 - 2 \cdot c_{CO,0} \cdot U_1} \quad (8)$$

$$X_{CH_4,1} = \frac{c_{CH_4,0} + c_{CO,0} \cdot U_1}{1 - 2 \cdot c_{CO,0} \cdot U_1} \quad (9)$$

$$X_{H_2O,1} = \frac{c_{H_2O,1} + c_{CO,0} \cdot U_1 - c_{CO,0} \cdot U_2}{1 - 2 \cdot c_{CO,0} \cdot U_1} \quad (10)$$

$$X_{CO,1} = \frac{c_{CO,0} - c_{CO,0} \cdot U_1 - c_{CO,0} \cdot U_2}{1 - 2 \cdot c_{CO,0} \cdot U_1} \quad (11)$$

$$X_{CO_2,1} = \frac{c_{CO_2,0} + c_{CO,0} \cdot U_2}{1 - 2 \cdot c_{CO,0} \cdot U_1} \quad (12)$$

$$X_{N_2,1} = \frac{c_{N_2,1}}{1 - 2 \cdot c_{CO,0} \cdot U_1} \quad (13)$$

$$U_{ges} = U_1 + U_2 = \frac{c_{CO,0} - c_{CO,1}}{c_{CO,0}} \quad (14)$$

In order to determine the chemical equilibrium at known temperature and pressure, a nonlinear equation system of the above mentioned equations need to be solved. The results of an example calculation are given in Figure 3.

The energy balance needs to be completed by the produced reaction heat which influences the equilibrium temperature. The amount of the produced reaction heat increases the conversion of the reactions is. A high reaction heat value leads to high temperatures.

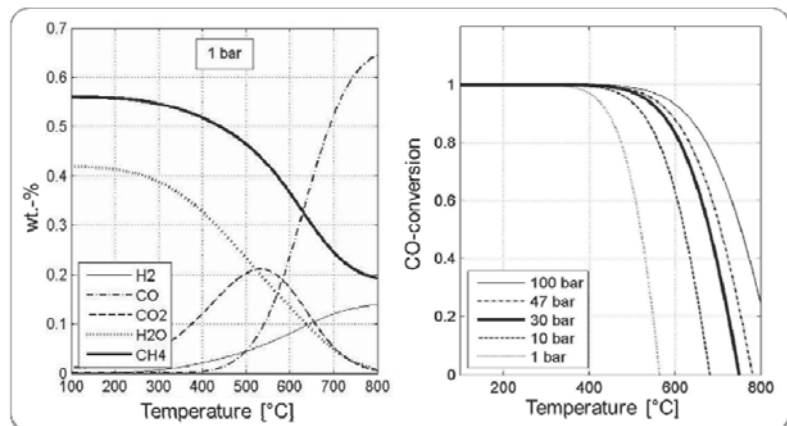


Figure 3. Chemical composition and CO-conversion at equilibrium state (initial composition in wt.-%: 33,5 H<sub>2</sub>; 6,7 CO; 3,7 CO<sub>2</sub>; 18,2 H<sub>2</sub>O; 37,4 CH<sub>4</sub>; 0,5 N<sub>2</sub>).

However, a high temperature leads to a low yield. The reaction heat flow,  $\dot{Q}_R$ , can be determined by equation 15, in which  $\Delta h$  is the specific reaction enthalpy of the reaction and  $M_{Gas}$  the mean molar mass of the gas. The temperature dependency of the specific reaction enthalpy is approximated with Equation 16. The parameters of equation 16 for reaction (R-1) and (R-3) are given in Table 3.

$$\dot{Q}_R = (\Delta h_{(R-1)} \cdot U_1 + \Delta h_{(R-3)} \cdot U_2) \cdot m_{Ges} \cdot (X_{CO,1} | M_{Gas,1} - X_{CO,0} | M_{Gas,0}) \quad (15)$$

$$\Delta h = A \cdot T^2 + B \cdot T + C \quad (16)$$

Reaction	A	B	C
(R-1)	0,0266	-47,7331	-205094,5788
(R-3)	0,0026	-7,4437	-41557,3842

Table 3. Parameters A, B and C of Equation 16 for reaction (R-1) and (R-3), [7].

The configuration of the reactor model is shown in Figure 4. The model components include a SNG\_reaction model, a volume, a heat capacity, a temperature sensor and a linear valve with a constant block. The calculation of the thermodynamic equilibrium occurs in the SNG\_reaction model. The volume component represents the volume of the reactor.

The heat capacity model is the heat capacity of the catalyst bed and the reactor wall. The valve model is used to simulate the pressure drop of the gas flow through the catalyst bed.

The reactor model has three main parameters: volume, pressure drop and heat capacity. The parameters have different impact on the reactor performance. The greater the reactor volume, the later the steady state will be achieved. The pressure drop changes the pressure and therewith the reactor yield. The pressure drop is predefined by the user. The impact on the heat capacity is more difficult. In the case of a high heat capacity, the reactor model needs to be split into a series of identical reactor zones, which are duplicates of the reactor model configuration as shown in Figure 4.

The higher the heat capacity, the more zones are needed to reach the steady state for temperature and methane at the same time, (which is physically necessary). If the number of zones was too low, the temperature change would not be in accordance with the thermodynamic equilibrium

An example of the transient temperature and methane concentration for a 5 zone reactor model at heat capacities in the range of 0 to 10 MJ/K is given in Figure 5. The cold reactor ignition or extinction can not be simulated, because no attention is paid to kinetic effects of the chemical reactions.

The steady state results of the simulation calculation are validated with a single reactor model of the TREMP methanization plant. The results are compared to two references.

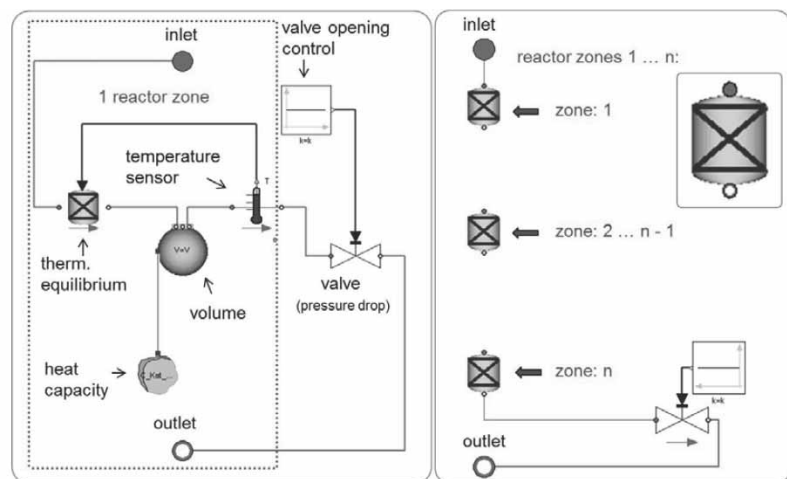


Figure 4. Reactor configuration (Left: 1 zone reactor; Right: multiple zone reactor).

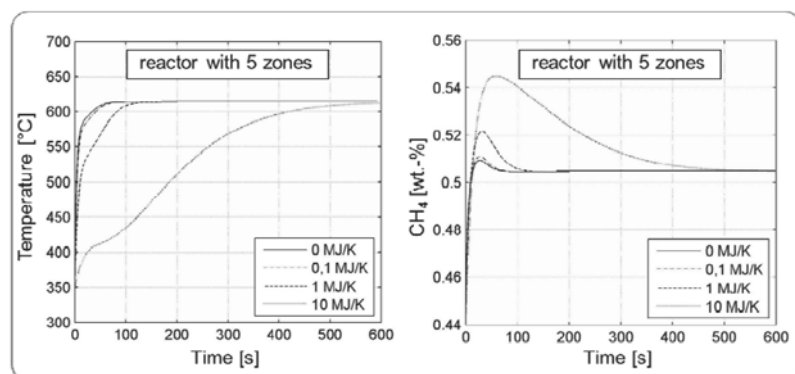


Figure 5. Temperature and methane concentration for a reactor model with 5 zones at heat capacities in the range of 0 to 10 MJ/K.

The first one is the experimental result of the TREMP which are published by Harms in [6]. The second reference is the steady state simulation result which is generated with the software ChemCAD by assuming the chemical equilibrium.

The results of the reactor model are compared with the data of these two references in Table 4. The error of the species is always lower than 4 % except for hydrogen at low concentrations, where the error can reach up to 10 % (under certain conditions).

**The Heat Exchanger Model**

The second important model for the modeling of the methanization plant is the heat exchanger, which is assumed by a simple design to enhance the numerical stability and the calculation speed. The heat exchanger design comprises an outside tube, filled with hot SNG gas, which has an inside tube filled with water. The configuration of a heat exchanger zone is given in the left image of Figure 6. The zone consists of two volume models. One volume model contains SNG gas and the other water/steam.

The volume model is connected by a thermal conduction model. The heat exchanger consists of series such zones. The water and the gas are in counter current flow as the fluid ports are set up in this way. A minimum number of zones are needed to reflect the phenomena that the outflow of the cold side can be hotter than the outflow of hot side, in certain cases. Consequently, a temperature profile can be generated along the heat exchanger as it is shown in the right image of Figure 6.

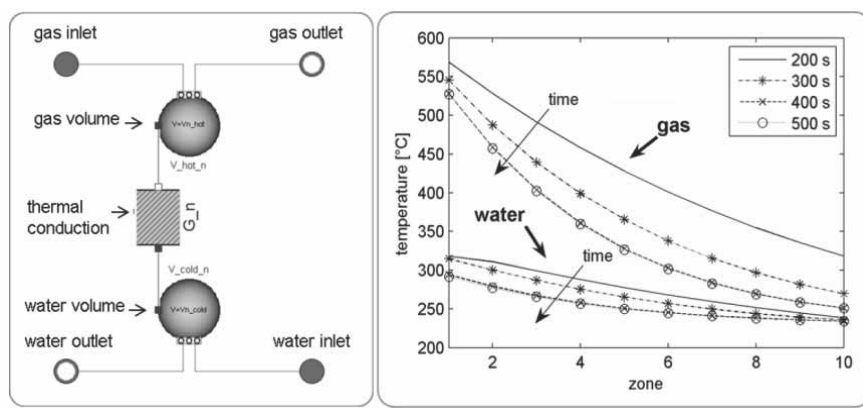


Figure 6. Left: Heat exchanger configuration for one zone; Right: Transient temperature profile of a heat exchanger model with 10 zones during a part load simulation.

	Unit	Inlet Tremp	Outlet Tremp	Outlet Modelica	Outlet ChemCAD
Temperature	°C	300	600	600	599
Pressure	bar	30	30	30	30
H <sub>2</sub>	wt.-%	4,97	2,47	2,40	2,41
CO	wt.-%	13,80	2,01	2,04	1,97
CO <sub>2</sub>	wt.-%	11,97	12,13	11,79	11,92
H <sub>2</sub> O	wt.-%	24,11	31,48	31,82	31,76
CH <sub>4</sub>	wt.-%	44,12	50,81	50,92	50,91
N <sub>2</sub>	wt.-%	1,03	1,10	1,03	1,03

Table 4. Comparison of the outlet gas composition of the first reactor of the TREMP with the reactor model in Modelica and in ChemCAD.

**3 Transient Part Load Behaviour**

The reactor and the heat exchanger model are used to build up the model of the methanization plant, as shown in Figure 2. The model of the methanization plant is utilized for transient simulation studies. All the simulations start with a transient oscillation as it is shown in the simulation example of the 1<sup>st</sup> reactor with recycle in Figure 7.

In the following, results of a part load simulation are presented. The simulation of the load change is faster than real time. In order to realize the part load, the feed mass stream was reduced to 50 % after reaching steady state (which is after 300 s simulation time) see Figure 8.

The steady state in part load is achieved after a simulation time of 1000 s. The results indicate lower gas temperatures in the reactors at part load leading to higher methane concentrations in the gas, see in Figure 9.

After 1000 s simulation time the feed mass stream was re-increased back to 100 % (full load). The steady state is reached after a simulation time of 2000 s. The temperature and the methane concentration achieve the same level as before the part load. Finally, the gas product quality at part load is almost equal to the qualities at full load. Furthermore, it is still possible to produce steam, but the pressure of the steam needs to be adjusted.



## 4 Conclusion

The paper shows that the dynamic modeling of a methanization plant is possible. Therefore simplifications are necessary which are implemented in the methanization plant model. The steady state results could be validated successfully by both a steady state flow sheet simulation with ChemCAD and technical process data from Harms [6]. Despite the simplifications in the model a good agreement with the technical process could be achieved. However, the dynamic behavior could be only validated by a plausibility check.

The methanization plant model is finally useful for the development and check of control concepts, and furthermore for the analysis of fault scenarios and load changes.

## References

- [1] Fahlke, J., Püschel, S., Hannemann, F., and Meyer, B. (2008). Modelling of the Gasification Island with Modelica. *Proc. 6<sup>th</sup> International Modelica Conference, March 2008*; Freiberg, Germany.
- [2] Heil, S., Brunhuber, C., Link, K., Kittel, J., and Meyer, B. (2009). Dynamic Modelling of CO<sub>2</sub>-removal units for an IGCC power plant. *Proc. 7<sup>th</sup> International Modelica Conference, Sept. 2009*, Freiberg, Germany.
- [3] Frohning, C., and Hammer, H. (1977). Methan (Erdgas-Austauschgas). In Falbe, Jürgen (Hrsg): *Chemierohstoffe aus Kohle*. Georg Thieme Verlag, Stuttgart.
- [4] Haldor Topsøe. (2010). *From solid fuels to substitute natural gas (SNG) using TREMP*. <http://www.topsøe.com>
- [5] Graf, F., and Bajohr, S. (2009). Erzeugung von SNG aus ligninreicher Biomasse. *Energie/Wasser-Praxis*, (4), 10-16.
- [6] Harms, H., Höhle, B., and Skov, A. (1980). Methanisierung kohlenmonoxidreicher Gase beim Energietransport. *Chemie.-Ing. Technik*, 6, 504-515.

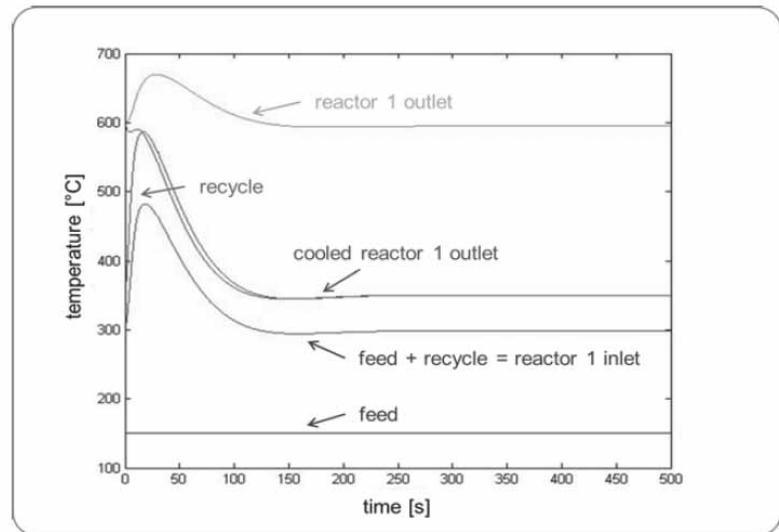


Figure 7. Transient oscillation of the gas temperatures during the start up.

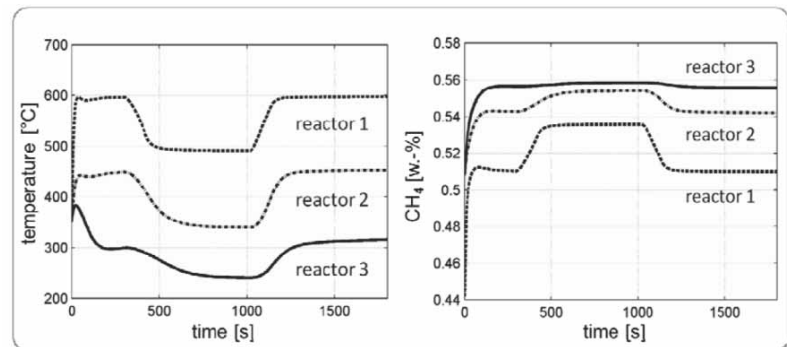


Figure 8. Transient gas temperatures and methane concentration at the outlet of reactor 1, 2 and 3 during at part load.

- [7] Factsage (2010). *Database FACT5*. From the Software FactSage, [www.factsage.com](http://www.factsage.com)
- [8] Anderlohr, A.: (1979) *Untersuchung zu gleichzeitigen Methanisierung und Konvertierung von CO-reicher Gase in einer katalytischen Wirbelschicht*. PhD Thesis, Fakultät für Chemieingenieurwesen, Tech. Hochschule Karlsruhe.

Submitted: September 2011 (ASIM STS Winterthur)

Accepted: July 20, 2012

# Dynamic Modeling and Simulation of Low Density Polyethylene Production – A Comparative Study

Ilknur Disli<sup>1</sup>, Achim Kienle<sup>1,2\*</sup>

<sup>1</sup>Institut für Automatisierungstechnik, Otto-von-Guericke-Universität Magdeburg, Postfach 4120, 39106 Magdeburg, Germany; \*[achim.kienle@e-technik.uni-magdeburg.de](mailto:achim.kienle@e-technik.uni-magdeburg.de)

<sup>2</sup>Max-Planck-Institut für Dynamik komplexer technischer Systeme, Magdeburg, Germany

**Abstract.** This study deals with the modelling and simulation of low density polyethylene (LDPE) production. LDPE is one of the most widely produced polymers which appears in the form of simple goods in our everyday life (e.g. food and pharmaceutical packaging, carrier bags). It is produced in a complex industrial process which takes place under extreme operating conditions (at pressure of ca. 2000-3000 bar) and may lead to nonlinear dynamics due to highly exothermic addition polymerization reactions. In principle, the process is represented by a distributed system with an external coordinate (the reactor length > 1000m) and various internal coordinates (the chain length of the polymer molecules, short and long chain branching and the number of double bonds), which can have strong effect on product properties. In this contribution a detailed reference model is introduced and possible model simplifications are discussed systematically from an on-line optimization and control point of view.

## Introduction

Low density polyethylene (LDPE) is one of the most widely produced polymers which appears in the form of simple goods in our everyday life (e.g. food and pharmaceutical packaging, carrier bags). It is produced in a complex industrial process which takes place under extreme operating conditions (at pressure between 2000-3000 bar). It can show intricate nonlinear dynamic behaviour due to the high exothermicity of the chemical reaction in combination with some internal recycle loops [7, 8], which requires some stabilizing control.

Usually, such a plant receives the raw materials from upstream processes which may cause load changes in the LDPE plant. It is important to maintain the product quality in the face of such load changes. Furthermore, in such a plant, typically more than 15 different grades of product with different product properties (density, melt flow index etc.) may be produced.

In order to follow frequently changing market demands it is necessary to follow an optimum grade transition policy during manufacturing different polymer while maintaining a profitable operation.

These issues emphasize the significance of control tasks for the LDPE plant and subsequently the significance of suitable process models which can be employed for control purposes. First attempts on optimization based control were based on steady state models only [4]. However more recently it is possible to use dynamic optimization methods within the framework of nonlinear model predictive control (NMPC) based on dynamic models. Although first steps have been made in this direction [9], the size of the mathematical model arising from the distributed nature of the LDPE plant remains a challenging issue.

The objective of this study is to provide a benchmark model of the LDPE plant with a reasonable model size which can be useful for optimal control approaches. Therefore first a detailed reference model developed in our group [2, 1] is reviewed and then possible model simplifications are introduced systematically. The reference model of the LDPE plant is represented by a distributed system with an external coordinate (the reactor length > 1000 m) and various internal coordinates (the chain length of the polymer molecules, short and long chain branching and the number of double bonds), which can have strong effect on product properties.

This model describes the heat transfer in detail in addition to including material recycles, both of which are shown to have influence on the plant dynamics. The experience gained on the process via reference model is used to make necessary model simplifications. The simulation with the suggested simple model are carried out to analyse the effects of heat transfer, the types of initiators and modifier on the state profiles and product properties. Results of the simple and reference models are compared and conclusions are drawn.

## 1 Process Description

The flow diagram of the process, which is shown in Figure 1, includes a tubular reactor, some peripheral units such as compressors, separators, heat exchangers, mixers, two recycle streams. The tubular reactor of ethylene polymerization (i.e. LDPE production) operates under high pressures between 2000 and 3000 bar. Reactor temperature along the reactor varies between 400 K and 600 K. Such a reactor is usually very long ( $> 1000$  m) and has a small diameter with a large ratio of length to diameter such as 25000.

The feed stream to the reactor consists of high purity ethylene as monomer and a suitable modifier (chain transfer agent). Most often a mixture of peroxides is used as initiator and fed at four injection points along the reactor. Feed stream to the reactor is first preheated then its pressure is raised to the required value through the compressors. The addition polymerization reactions shown in Table 1 are initiated by decomposition of initiators around the injection positions. Typically ethylene polymerization is highly exothermic such that the heat released during the process is required to be removed partially by cooling jackets. Right after each initiator injection point there exist four cooling jackets.

Furthermore, in order to keep the reactor temperature below the allowed maximum value, the conversion in the reactor is kept low by adjusting the initiator flow rates in addition to using the available cooling capacity. The polymer product, LDPE, is withdrawn from the last separator whereas the unreacted ethylene and modifier are recycled from both separators to the inlet of the reactor to be mixed with the fresh feed.

## 2 Reference Model

A detailed reference dynamic model of the LDPE production plant has been developed in our group [2, 1]. The process is represented by a distributed system with an external coordinate (the reactor length of 2 km) and various internal coordinates (the chain length of the polymer molecules, short and long chain branching and the number of double bonds), which can have strong effect on product properties.

In the reactor it is assumed that: (i) ethylene-polyethylene mixture is homogeneous; (ii) there exist only liquid phase flowing in plug-flow mode without axial mixing; (iii) physical properties are function of temperature, pressure and composition; (iv) quasi-stationary pressure dynamics is valid; (v) a detailed

reaction mechanism is considered. In peripheral units quasi-stationary energy balance and well mixed fluids are assumed. The chemical reactions of free radical addition polymerization are shown in Table 1.

The polymer product, LDPE, includes polymer molecules of different chain lengths. It is necessary that the chain length distribution of the product should be described. For this purpose, the method of moments is used by including only the first 3 moment balances of dead and live polymer molecules.

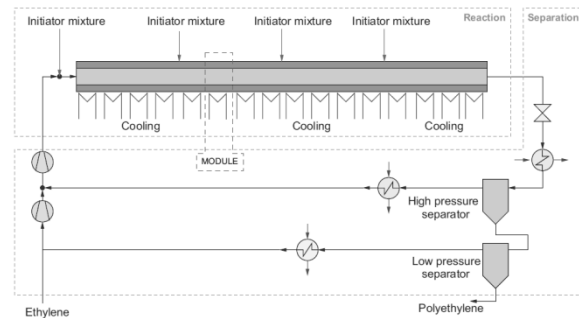


Figure 1. Flow diagram of LDPE plant.

The tubular reactor is represented by 16 modules in series. Each module comprises a part of reactor and a countercurrent coolant cycle around it. The model describes the heat transfer in detail in addition to including material recycles, both of which are shown to have influence on the plant dynamics. Model equations based on momentum, mass and energy balances are summarized for one reactor module as follows:

Reactor:

$$\frac{\partial x_i}{\partial t} + v(z, t) \frac{\partial x_i}{\partial z} = \sum_{j=1}^J v_{ij} r_j \quad 0 < z < L, i = 1, \dots, NC \quad (1)$$

$$\rho c_p \left( \frac{\partial T}{\partial t} + v(z, t) \frac{\partial T}{\partial z} \right) = \sum_{j=1}^J (\Delta H_r)_j r_j - U_{wi}(T - T_w) \quad (2)$$

$$\frac{\partial P}{\partial z} = -\frac{1}{2} \zeta \frac{\dot{m}^2}{\rho(z, t) d} \quad (3)$$

$$\text{Wall: } \rho_w c_{pw} \left( \frac{\partial T_w}{\partial t} + \beta(z, t) \frac{\partial T_w}{\partial z} \right) = U_{wi}(T - T_w) - U_{wo}(T_w - T_c) \quad (4)$$

$$\text{Coolant cycle: } \rho_c c_{pc} \left( \frac{\partial T_c}{\partial t} + \beta(z, t) \frac{\partial T_c}{\partial z} \right) = U_{wo}(T_w - T_c) - U_{ex}(T_c - T_{amb}) \quad (5)$$

There  $z$  is the axial coordinate;  $L$  is the total length of a reactor section and  $t$  is the time;  $v$  is the axial velocity of the reaction fluid;  $v$  is the stoichiometric coefficient;  $j$  is the index for reaction;  $r$  is the rate of reaction;  $\Delta H_r$  is the heat of reaction;  $\dot{m}$  is the mass flow through the cross section of the tubular reactor with diameter  $d$ ;  $\rho$ ,  $C_p$ ,  $U$  are density, heat capacity and heat transfer coefficient at different layers.

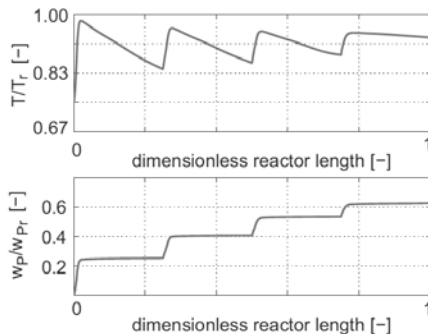
Main reactions		Side reactions	
Initiation		Chain transfer	
<i>Initiator decomposition</i>	$I_2 \xrightarrow{k_d} 2I^*$	<i>to Monomer</i>	$R_n + M \xrightarrow{k_{tr,M}} P_n + R_1$
<i>Chemical initiation</i>	$I^* + M \xrightarrow{k_i} R_1$	<i>to Polymer</i>	$R_n + P_m \xrightarrow{k_{tr,P}} P_n + R_{sec,m}$
<i>Thermal initiation</i>	$3M \xrightarrow{k_{th}} 2R_1 + M$	<i>to Modifier</i>	$R_n + X \xrightarrow{k_{tr,X}} P_n + R_X$
Propagation		Initiation	$R_X + M \xrightarrow{k_{p,X}} R_1$
Termination	$R_n + M \xrightarrow{k_p} R_{n+1}$	Back biting	$R_n \xrightarrow{k_{bb}} R_n$
<i>by Combination</i>	$R_n \xrightarrow{k_{bb}} R_n$	Propagation of secondary radicals	$R_{sec,n} + M \xrightarrow{k_{p,sec}} R_{n+1}$
<i>by Disproportionation</i>	$R_{sec,n} + M \xrightarrow{k_{p,sec}} R_{n+1}$	$\beta$ -scission	$R_{sec,n} \xrightarrow{k_\beta} P_{n-k} + R_k$

**Table 1:** Reaction mechanism :  $I$  represents the initiator,  $I^*$  is the initiator radical,  $M$  is the monomer,  $R$  is the growing or live polymer radical,  $P$  is the dead polymer,  $n$  and  $m$  denote the degree of polymerization.

The state vector  $x_i$  represents weight fraction of monomer, initiators, modifier, and related radicals, moments of living and dead polymer chains. The state vectors  $T$ ,  $T_w$ ,  $T_c$ ,  $T_{amb}$ , represent temperatures of reaction fluid, reactor wall, cooling medium and ambient air respectively.

The balance equations for one reactor module result in a set of 30 partial differential and algebraic equations (PDAEs). The peripheral units are represented by differential algebraic equation systems (DAE). The simulation of the dynamic behaviour of the process is done within the flow-sheet simulation package DIVA [6]. As a priority to that the PDAE system is transformed to a DAE system by using the adaptive method of line with finite difference scheme. After discretization of the spatial coordinate (60 grid points for one module) the final DAE system consists of approximately 30000 dynamic and state variables [1].

This model was validated successfully by showing very good agreement with steady state data available from industrial partners. The steady state temperature and conversion profiles are shown Figure 2.



**Figure 2.** Detailed model - Dimensionless steady state temperature and conversion profiles along the reactor.

Hafele *et al.* [1] further studied the influence of the heat transfer through reactor wall and the influence of material recycles on the plant dynamics. It has been shown that: (i) the heat transfer through thick reactor wall dominates the time constant of the reactor; (ii) by including the material recycles the time constant is significantly increased and in some cases they give rise to intricate non-linear behaviour.

In general, the detailed model represents the steady state and dynamic behaviour of the industrial LDPE plant very well.

However the size of the model is required to be reduced in order to carry out on-line optimization and control studies within a reasonable computation time. Therefore in the following section a simplified version is proposed.

### 3 Simple Model

The simple model of the LDPE plant is based on the mass and energy balances. Main simplifications are done on the plant flow-diagram, the reaction mechanism and the energy balance in addition to some other model assumptions.

**Simplified plant flow-diagram.** The simplified flow diagram of the LDPE plant is shown in Figure 3. It comprises a tubular reactor with four modules in series, one flash unit and one recycle stream. The operating conditions are the same as in the case of detailed model.

**Simplified reaction mechanism.** The reaction mechanism of the simple model includes the main reactions which are highlighted in grey colour in Table 1 whereas the detailed plant model considers all reactions listed there. Kinetic parameters are given in Table 2 .

**Simplified energy balance.** It is assumed that (i) the overall heat transfer coefficient  $U$  has a constant value in each reactor module ; (ii) the cooling temperature  $T_c$  is constant along the whole cooling jacket. These lead to neglecting the energy balance equations on the coolant medium and the reactor wall side.

Additionally, the following assumptions hold: one phase (liquid) flow; no axial dispersion (plug flow); constant pressure drop; constant physical properties; negligible time delay between peripheral units; constant temperature in the recycle; ideal separation in the flash unit. Mass and energy balances for both flash and mixer units are assumed to be quasi-stationary.

### 3.1 Model Equations

The model equations for one reactor module, which comprises a part of the reactor between two initiator injection points and one counter current coolant cycle around it, are given as follows:

$$\frac{\partial C_i}{\partial t} + v \frac{\partial C_i}{\partial z} = \sum_{j=1}^J v_{ij} r_j \quad 0 < z < L, i = 1, \dots, NC \quad (6)$$

$$\rho C_p \left( \frac{\partial T}{\partial t} + v \frac{\partial T}{\partial z} \right) = \sum_{j=1}^J (\Delta H_r)_j r_j - U(T - T_c) \quad (7)$$

The boundary conditions:

$$C_i(0, t) = C_{i, in}(t) \quad T(0, t) = T_{in}(t)$$

The initial conditions:

$$C_i(z, 0) = C_{i, 0}(z) \quad T(0, t) = T_{in}(t)$$

The state vector  $C_i$  represents concentrations of monomer, initiators, modifier and related radicals, moments of living and dead polymer chains. The state vector  $T$  represents the reactor temperature. The method of moments is used to represent the progress of the reaction in terms of the leading moments of the chain length distribution of the 'live' and 'dead' polymer chains. These moments are defined by the following equations:

$$\lambda_i = \sum_k^{\infty} k^i [R_k], \quad \mu_i = \sum_{k=2}^{\infty} k^i [P_k] \quad (8)$$

where  $\lambda_i$  and  $\mu_i$  are the  $i^{th}$  moments of the living polymer and the dead polymer chains respectively.

$0^{th}$  moment corresponds to the total concentration of the polymer.  $1^{st}$  and  $2^{nd}$  moments are used to characterize the molecular weight distribution, e.g. the number average chain length, NACL, also known as degree of polymerization,  $DP_n$ , and the polydispersity,  $PD$ :

$$NACL = \mu_1 / \mu_0 \quad (9)$$

$$PD = (\mu_2 \mu_0) / (\mu_1)^2 \quad (10)$$

If one is interested only in the stability control of the plant then the reactor model with  $0^{th}$  moment is satisfactory. On the other hand, the product quality information (e.g. NACL and PD) is crucial for grade transition control problems.

It requires to include at least 3 leading moments ( $0^{th}$ ,  $1^{st}$  and  $2^{nd}$ ) in the model equations. Accordingly the size of the model will increase.

The simple model formulation (with 3 leading moments) consists of a set of 14 PDAE per reactor module. After transforming this PDAE system into a DAE system, simulations are carried out within the flow-sheet simulation package DIVA [6]. The results of the simple model are presented in the following section.

### 3.2 Results

Results obtained from simulations of the simple model are presented for 4 different cases.

In Case I, II and III, the effects of the overall heat transfer coefficient and the types of initiators on steady state profiles are shown. In these cases the reaction mechanism includes only the main reactions listed in Table 1. Case IV includes additionally the reaction related to the modifier consumption and aims to show the influence of modifier on product properties. The results obtained in each case are compared to those of the reference model.

#### Case I.

In Case I, it is assumed that: (i) one type of initiator with a constant flow-rate is fed at each injection point; (ii) the heat transfer coefficient  $U$  has the same constant value in all four modules. The steady state profiles of temperature and concentration along the reactor are illustrated in Figure 4. The reference model profiles are represented by red solid lines, the state profiles of the simple model-Case I by blue dashed-dotted lines. Results of the Case I are qualitatively in agreement with the detailed model but there exists a large deviation from reference state profiles.

This is not unexpected due to many simplifying assumptions made. In the next step these assumptions will be modified by considering the behaviour of the reference model.

#### Case II

In Case II, the effect of the heat transfer coefficient  $U$  on the steady state profiles is shown. The detailed model results indicate that  $U$  decreases along the reactor. Consequently, for the simple model,  $U$  will be specified at a different mean value in each reactor module but in a decreasing manner through the end of the reactor. The assumption of using one type of initiator in each module is still valid. The resulting profiles of Case II are illustrated in Figure 4 by pink dashed lines. It can be noted

that some improvement is obtained for the temperature profile of the simple model especially through the end of the reactor. However, a considerable large offset still remains between the states of simple and reference models.

**Case III**

In Case III, a mixture of two types of initiators is used at each injection point, which is a more realistic approach. Additionally kinetic parameters of initiator decomposition reaction are taken from the reference model. The condition on the parameter *U* is the same as in Case II: *U* is kept constant at a different value in each reactor module. The results of Case III are demonstrated in Figure 4 by black solid lines. It is shown that almost perfect agreement is obtained quantitatively between the state profiles of simple and reference models.

At this point, a question arises about the product quality. Conversion, NACL and PD, are given in Table 3. Although the ethylene conversion approximately agrees for the reference model and the simple model (Case-III), there is a considerable deviation in NACL and PD values. This is due to neglecting all side reactions of the reaction mechanism up to this point. Side reactions have functions to stop a growing polymer chain or to cause branching along a polymer chain etc. In the next step, the most important side reaction will be consider in the reaction mechanism of the simple model.

**Case IV**

In Case IV, the conditions are the same as in Case III except that the reaction mechanism is extended by including the chain transfer reaction to the modifier which is responsible to regulate the chain length of the polymer molecules. The state profiles obtained in Case IV are the same as in Case III. Moreover, NACL and PD values are improved considerably (see Table 3). Especially NACL with a reative value of 0.99 is very close to that of thereference model.

	Conversion	NACL	Polydispersity
Simple model (Case III) (no modifier)	0.876	17.18	2.81
Simple model (Case IV) (With modifier)	0.876	0.99	0.49

Table 3. Product quality values relative to the reference model

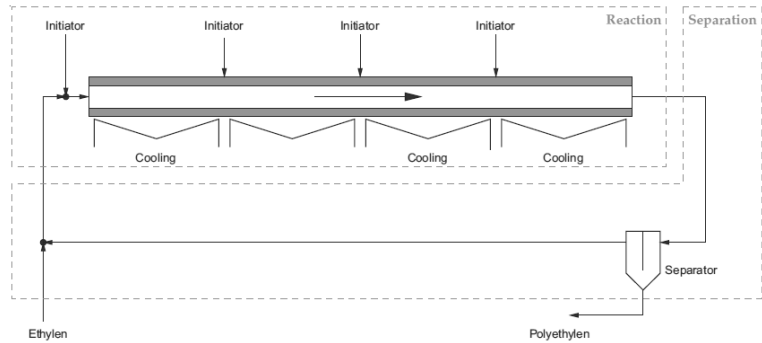


Figure 3. Simplified flow diagram of LDPE plant.

Initiation		
<i>Init. decomposition</i>		
$I_2 \xrightarrow{k_d} 2I^*$	$k_{dL} = 1.35 \times 10^{13} \exp\left(\frac{-117476 - 0.2805 \times 10^{-5} P}{R_g T}\right)^a$	$[m^3/s]$
	$k_{dH} = 2.89 \times 10^{14} \exp\left(\frac{-138237 - 1.012 \times 10^{-5} P}{R_g T}\right)^a$	$[m^3/s]$
<i>Chemical initiation</i>		
$I^* + M \xrightarrow{k_1} R_1$	$k_1 = k_p$	
Propagation		
$R_n + M \xrightarrow{k_p} R_{n+1}$	$k_p = 5.88 \times 10^4 \exp\left(\frac{-29704 + 2.325 \times 10^{-5} P}{R_g T}\right)^b$	$[\frac{m^3}{kmol \cdot s}]$
Termination		
<i>by Combination</i>		
$R_n + R_m \xrightarrow{k_{tc}} P_{n+m}$	$k_{tc} = 1.075 \times 10^6 \exp\left(\frac{-1247 + 1.422 \times 10^{-5} P}{R_g T}\right)^b$	$[\frac{m^3}{kmol \cdot s}]$
<i>by Disproportionation</i>		
$R_n + R_m \xrightarrow{k_{td}} P_n + P_m$	$k_{td} = k_{tc}$	

Table 2. Kinetic rate constants: <sup>a</sup> Kim and Iedema [3], <sup>b</sup> Lee and Marano [5]

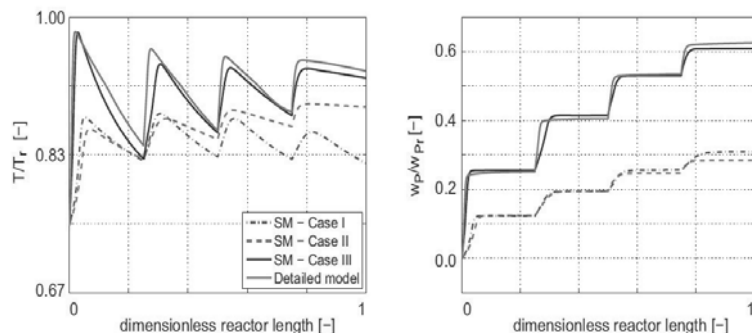


Figure 4. Dimensionless steady state temperature and conversion profiles along the reactor : Red continuous line - Detailed model; Dashed line - Simple model-I; Dashed and dotted line- Simple model-II; Black continuous line – Simple model-III.

As a result of these four case studies, it can be concluded that the simple model with the conditions applied in Case IV represents the LDPE production process reasonably well close to the reference states and product properties. It can be used in principle research for grade transition.

#### Model Sizes.

There may be some differences in the model sizes among the cases studied above. This depends on (i) the number of initiators - whether one type or a mixture of few types of initiators is used ; (ii) whether the modifier reaction is included or not. (iii) the number of leading moments included. However, any version of the simple models has a much smaller size than the reference model. The comparison of the model sizes for reference and simple models are given in Table 4. The order of the ODAE system for the simple model-Case IV is 5600 which is reduced by a factor of 5 comparing to the order of 30000 in the case of reference model. If one is interested in only stability control of the plant, then one should use a simple model with only 0<sup>th</sup> moments, which will result in a model with an order of 2400.

	Reference Model [2,1]	Simple model Case IV	Simple model 0 <sup>th</sup> moment
Number of reactor module	16	4	4
Order of PDAE sys	30 / module	14 / module	6
Equidistant grid points	60 / module	100 / module	100 / module
Order of the ODAE sys	30000	5600	2400

Table 4. Comparison of model sizes.

## 4 Conclusions

In this study a detailed reference model was introduced and possible model simplifications were discussed systematically from an on-line optimization and control point of view. The simulation with the suggested simple model were carried out to analyse the effects of the overall heat transfer coefficient, the types of initiators and modifier on the state profiles and product properties. The simple model was validated by comparing with the reference model. It was shown that the simple model

which has a much smaller size than reference model works fine except some small deviations and has potential for stabilizing control studies. Considering that it also provides product properties at least in a qualitative way, the simple model can be used in principle research for grade transition control.

#### References

- [1] M. Hafele, M., Kienle, A., Boll, M., and Schmidt, C.-U. (2006). Modeling and analysis of a plant for the production of low-density polyethylene. *Comput. Chem. Engng.*, 31, 51-65.
- [2] M. Hafele, M., Kienle, A., Boll, M., Schmidt, C.-U., and Schwibach, M. (2005). Dynamic simulation of a tubular reactor for the production of low-density polyethylene using adaptive method of lines. *J. Computational and Appl. Math.*, 183(2), 288-300.
- [3] Kim, D.-M., and Iedema, P.(2004). Molecular weight distribution in low-density polyethylene polymerization ; impact of scission mechanism in the case of a tubular reactor. *Chem. Eng. Sci.*, 59, 2039-2052.
- [4] Kiparissides, C., Verros, G., and MacGregor, J. (1993). Mathematical modeling, optimization and quality control of high-pressure ethylene polymerization reactors. *J. of Macromol. Sci.-Rev. Macromol.Chem. Phys.*, C33(4), 437-527.
- [5] Lee, K., and Marano, J. (1979). Free-radical polymerization: sensitivity of conversion and molecularweights to reaction conditions. *ACS Symp.Ser.*, 104, 221-251.
- [6] Mangold, M., Kienle, A., Mohl, K., and Gilles, E.(2000). Nonlinear computation in diva - methods and applications. *Chem. Eng. Sci.*, 55, 441-454.
- [7] Pushpavanam, S., and Kienle, A. (2001). Nonlinear behavior of an ideal reactor separator network with mass recycle. *Chem. Eng. Sci.*, 56, 2837-2849.
- [8] Ray, W., and Villa, C. (2000). Nonlinear dynamics found in polymerization processes - a review. *Chem. Eng. Sci.*, 55, 275-290.
- [9] Zavala, V., and Villa, C. (2008). Large-scale nonlinear programming strategies for the operation of low-density polyethylene tubular reactors. In Proc. *ESCAPE-18, Lyon, France*, 629-634.

Submitted MATHMOD 2009: November 2008

Accepted MATHMOD 2009: January 2009

Submitted SNE: January 2010

Accepted: February 5, 2010



# A Software Framework for Modeling and Simulation of Dynamic Metabolic and Isotopic Systems

Jana Tillack<sup>1</sup>, Stephan Noack<sup>1\*</sup>, Katharina Nöh<sup>1</sup>, Atya Elsheikh<sup>2</sup>, Wolfgang Wiechert<sup>2</sup>

<sup>1</sup> Inst. of Biotechnology, Research Centre Jülich, Leo-Brandt-Strasse, 52425 Jülich, Germany; \*[s.noack@fz-juelich.de](mailto:s.noack@fz-juelich.de)

<sup>2</sup> Inst. of System Engineering., Dept. of Simulation, University of Siegen, Germany

**Abstract.** Various experimental approaches including also isotopic tracer information were proposed paving the way for quantitative modeling and detailed *in vivo* studies of biological systems. Accordingly, for the underlying modeling approaches a diversity of *in silico* tools have been developed. The full exploitation of this potential to address metabolic processes is hampered by mainly three principal issues.

First, not all currently realizable experiments are covered by these tools. Secondly, an easy switching between existing tools allowing for a flexible description of different experimental states is not possible. The third item addresses the universality of underlying modeling concepts which usually have a restricted focus.

We propose a general modeling concept which allows modeling and simulation of all combinations of metabolically and isotopically variants in their stationary and dynamic states and which is embedded in an unique software platform.

The basic idea is to build up dynamic metabolic networks relying on mass balances for intermediate labeling pools. A workflow is presented that allows the automatized generation of models of any size and complexity specially tailored for the experiment of choice. Within the software framework, the application of sophisticated methods for statistical analysis and interpretation of simulation results are realized.

## Introduction

To reveal underlying kinetic mechanisms of metabolic regulation, pulse experiments have been established to generate dynamic data of metabolic intermediates [11, 7]. For the description of these data sets, dynamic metabolic networks are formulated that are based on kinetic models describing enzyme catalysis and regulatory metabolic interactions [17].

Validation of such models is a challenging task since usually a huge amount of parameters have to be identified with only a limited number of measurements available.

In order to attenuate the solution of these usually ill-posed problems, but nevertheless being able to elucidate metabolic stationary phenotypic behavior, metabolic flux analysis (MFA) has been introduced [14].

This approach has been refined by the addition of isotopically labeled tracers (usually <sup>13</sup>C-labeled glucose) [19]. In the last decade <sup>13</sup>C-MFA has become one of the major tools of metabolic engineering which is successfully applied to gain biological insight into different organisms of bacteria and plants [10, 13]. However, because it operates under metabolic stationary conditions, <sup>13</sup>C-MFA is not capable of describing *in vivo* metabolic regulation and control. Therefore its predictive power is limited.

Ongoing development of experimental and analytical procedures for measuring metabolic intermediates with and without tracer information (e.g. <sup>13</sup>C) [12] led to the requirement to formulate different model approaches describing the measurement data. Similar to the previous step from MFA to <sup>13</sup>C-MFA recently a supplementation of the dynamic metabolic modeling approach with <sup>13</sup>C labeling has been suggested in [16]. Taking all approaches together a classification can be derived between the model's assumption on metabolic and isotopic (non-)stationarity.

In the following we present a general framework for modeling and simulation of all common types of metabolic and isotopic systems.

In order to clarify our concept and introduce necessary definitions we start with a mathematical overview on the established model approaches which are introduced via a consistent example network.

# 1 Modeling Metabolic Networks

## 1.1 General Model Assumptions

Our modeling approach relies on the following assumptions:

- Continuum:** All chemical species involved in the considered processes have such a high copy number that a continuous concentration value can be used to describe it.
- Homogeneity:** Diffusion processes are very fast compared to chemical reactions so that concentrations can be considered to be spatially homogeneous.
- No isotope effects:** There are no significant isotopic mass effects, i.e. the reaction rates do not depend on the actual labeling state of the reactants.

With these assumptions, which are common for most of the currently available averaging biochemical network concepts [19], it is possible to describe reaction networks with metabolite pools and metabolic fluxes as state variables.

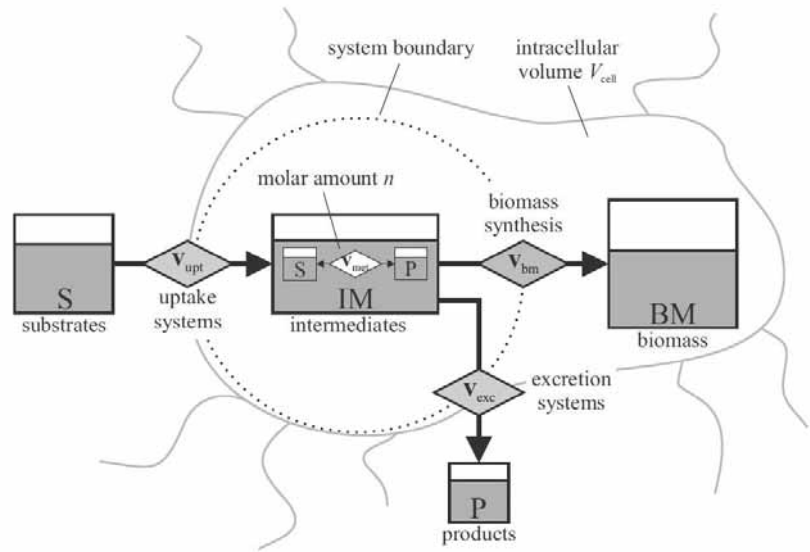
## 1.2 Mass Balances of Intermediate Pools

In principle, the cell's fluxome can be divided into two different species (cf. Figure 1):

- Extracellular fluxes, comprising uptake systems ( $v_{upt}$ ) for various substrates ( $S$ ) into the cell and excretion systems ( $v_{exc}$ ) for products ( $P$ ) out of the cell.
- Intracellular fluxes, comprising reactions ( $v_{met}$ ) between metabolic intermediates ( $IM$ ) and effluxes  $v_{bm}$  from intermediate precursors into biomass components ( $BM$ ).

Following this classification a mass balance for the molar amount  $n$  of each metabolic intermediate in the cell can be formulated as:

$$\frac{dn_{IM}}{dt} = \sum v_{upt} - \sum v_{exc} \pm \sum v_{met} - \sum v_{bm} \quad (1)$$



**Figure 1.** Scheme of metabolic processes within a cell that are covered by metabolic network models. Pools within the cell wall (indicated by the grey line) are mass balanced. The cell is fed with substrates and metabolizes intermediates towards biomass and excretion products.

By introducing (a) the relation  $c_{IM} = n_{IM}/V_{cell}$  and (b) assuming that the intracellular volume is constant  $\frac{dV_{cell}}{dt} = 0$ , Equation 1 can be transformed into a mass balance for intermediate concentrations  $c_{IM}$ :

$$\begin{aligned} \frac{dn_{IM}}{dt} &\stackrel{(a)}{=} \frac{d(c_{IM} \cdot V_{cell})}{dt} = \\ &= c_{IM} \frac{dV_{cell}}{dt} + V_{cell} \frac{dc_{IM}}{dt} \stackrel{(b)}{=} V_{cell} \frac{dc_{IM}}{dt} \end{aligned} \quad (2,3)$$

For a simple example shown in Figure 2 the time dependent concentration changes of all intracellular metabolites are formulated as:

$$\begin{aligned} \frac{dc_A}{dt} &= v_{upt} - v_1 & \frac{dc_B}{dt} &= v_1 - v_2 - v_5 \\ \frac{dc_C}{dt} &= v_2 - v_3 & \frac{dc_D}{dt} &= v_3 - v_4 \\ \frac{dc_E}{dt} &= v_4 + v_5 - v_{bm} & \frac{dc_F}{dt} &= v_3 + v_4 - v_{ex} \end{aligned} \quad (4)$$

The system can be written in matrix notation:

$$\frac{d\mathbf{c}_{IM}}{dt} = \mathbf{N} \cdot \mathbf{v} \quad (5)$$

with vectors for all intermediates  $\mathbf{c} = (c_A, c_B, \dots, c_F)^T$  and fluxes  $\mathbf{v} = (v_1, v_2, \dots, v_{ex})^T$ .  $\mathbf{N}$  is called stoichiometric matrix, which consists of  $m$  columns representing all reactions and  $n$  rows representing all balanced IM's.

**Metabolic Stationary States.** Metabolic stationary flux analysis (MFA) aims at the quantification of intracellular fluxes in the metabolism of an organism. In order to use this approach, the assumption on metabolic stationary  $dc_{IM}/dt = 0$  must hold, i.e. the concentrations of all IM's do not change over the experimental time. It is generally accepted that this condition is fulfilled in bioreactors under continuous cultivation conditions (chemostat) and in the exponential growth phase of cells cultivated in batch/fed-batch mode.

Assuming a metabolic stationary system state it simplifies as follows according to Equation 5:

$$0 = \mathbf{N} \cdot \mathbf{v} = \mathbf{f}(\mathbf{v}) \quad (6)$$

Due to the fact that there are usually more reactions than metabolite pools, the algebraic equation system 6 is underdetermined. Including measurements (typically substrate uptake and product excretion rates determined from extracellular metabolome analysis as well as anabolic reactions known from biomass formation) is usually not enough to recover all flux rates. Thus, only a linear combination of all fluxes can be calculated. In the running example a combination of all intracellular fluxes can be determined by constraining the influx  $v_{feed}$  and the two effluxes  $v_{bm}$  and  $v_{ex}$ .

**Metabolic Non-stationary States.** Strictly speaking a biochemical system under defined cultivation conditions can only approximatively be considered as metabolic stationary, i.e. usually a quasi-stationary state  $dc_{IM}/dt \approx 0$  is attained.

To be more general, dynamic metabolic networks can be formulated (cf. Equation 5):

$$\begin{aligned} \frac{d\mathbf{c}_{IM}}{dt} &= \mathbf{N} \cdot \mathbf{v}(\mathbf{c}_{IM}, \mathbf{c}_S, \alpha) = \\ &= f(\mathbf{c}_{IM}, \mathbf{c}_S, \alpha), \quad (7) \\ \mathbf{c}_{IM}(0) &= \mathbf{c}_{IM,0} \end{aligned}$$

with external (possibly time dependent) concentrations  $\mathbf{c}_S$  and the vector of kinetic parameters. Usually, the time dependent reaction rates  $\mathbf{v}$  are modelled mechanistically assuming fast equilibrium of intermediate enzyme complexes (validity of Michaelis-Menten Quasi-Steady State). For example the reaction rate  $v_1$  of the example can be modelled by a reversible reaction of Michaelis-Menten type

$$v_1 = v_{max,1} \cdot \frac{c_A - \frac{c_B}{K_{eq,1}}}{K_{A,1} \cdot \left(1 + \frac{c_B}{K_{B,1}}\right) + c_A} \quad (8)$$

where kinetic parameters are the maximal reaction rate  $v_{max}$ , the equilibrium constant  $K_{eq}$ , and the affinity constants  $K_A$  and  $K_B$ .

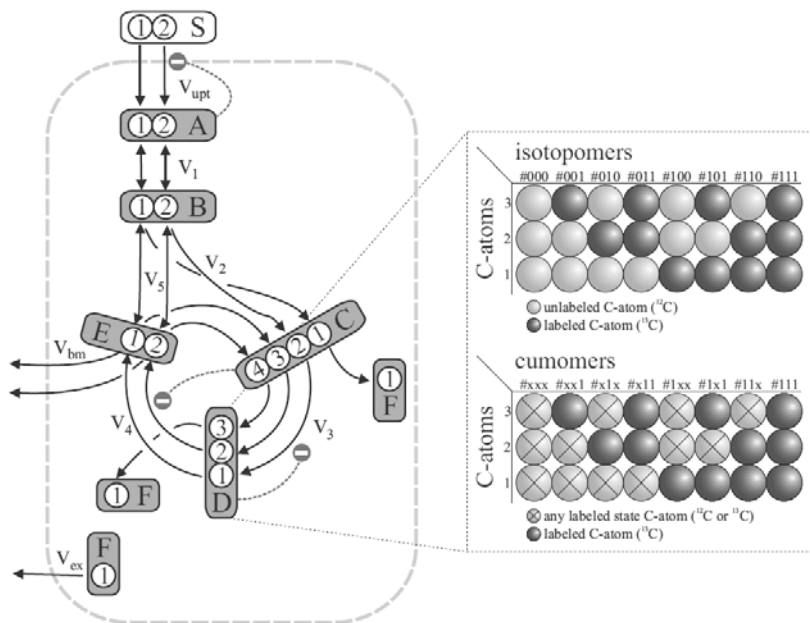


Figure 2. Left: Different levels of the example network regarded throughout the text. Level 1 (orange boxes): metabolic network model describing the reaction of rates  $v$  between intracellular metabolite pools. Level 2 (white cycles): C-atom transition network describing the carbon atom traces. Level 3 (red interactions): regulatory/inhibitory constraints. Right: Two different ways to express the isotopic composition of metabolite pools: isotopomers (isotopic isomers) and cumomers (cumulative isotopomers).

As a result of the mechanistic enzyme descriptions, dynamic models contain a high number of model parameters. Although rapidly sampled measurement data of time dependent concentration changes are used for parameter fitting, often the adequate information is missing for identifying all kinetic parameters [2]. To reduce the amount of unknown model parameters, several methods for simplifying enzyme kinetics have been proposed [5, 4]. Inserting mechanistic or approximative kinetic approaches in Equation 4 results in the ordinary differential equation (ODE) system for the running example under metabolic dynamic conditions.

### 1.3 Mass Balance of Labeled Intermediate Pools

The application of cultivation experiments with labeled substrates is motivated by the better relation between measurement data and model unknowns. Depending on the mathematical formalism for modeling the labeling state of a certain metabolic network this results in a combinatorial blowup of the number of equations [18, 15, 1].

Clearly, the most general representation of isotopic systems is given by the concepts of isotopomers and, equivalently, cumomers (cf. Figure 2). In both cases there is not only one single mass balance for the overall concentration of an IM, but rather  $2^n$  equations for the IM's isotopomer or cumomer concentrations, respectively, with  $n$  the number of IM's C-atoms.

Following the well-known isotopomer concept the mass balance for intermediate labeling pools  $\mathbf{c}_{IM}^{Iso}$  is formulated

$$\begin{aligned} \frac{d\mathbf{c}_{IM}^{Iso}}{dt} &= \frac{d(c_{IM} \cdot \mathbf{x}_{IM}^{Iso})}{dt} = c_{IM} \frac{d\mathbf{x}_{IM}^{Iso}}{dt} + \mathbf{x}_{IM}^{Iso} \frac{dc_{IM}}{dt} \\ \frac{dc_{IM}}{dt} &= \sum_{j=1}^n \frac{dc_{IM}^{Iso}}{dt} \end{aligned} \quad (9)$$

There  $x_{IM}^{Iso} = x_{IM}^{Iso}/c_{IM}$  denotes an isotopomer fraction of the intermediate IM.

When modeling systems where isotopic labeling plays a role, it is necessary to formulate mass balances around single intermediates that are based on a separation of the net rates  $v^{net} = v^{\rightarrow} - v^{\leftarrow}$  into forward  $v^{\rightarrow}$  and backward rates  $v^{\leftarrow}$ . The reason is simply given by the fact that a IM's labeling is influenced by the labeling fraction of all pools contributing to its (isotope) mass balance. Consequently, in a reversible reaction step, the labeling that arrives the substrate pool through the backward rate depends on the labeling of the product pool

and is not necessarily consistent with the labeling state of the substrate, except high exchange rates can be assumed.

As an example, consider the isotopomer mass balance around the labeled pool B of the example (cf. Figure 2):

$$\begin{aligned} \frac{d\mathbf{c}_B^{#ij}}{dt} &= v_1^{\rightarrow} \cdot \mathbf{x}_A^{#ij} + v_5^{\leftarrow} \cdot \mathbf{x}_E^{#ij} - \\ &-(v_1^{\leftarrow} + v_2^{\rightarrow} + v_5^{\rightarrow}) \cdot \mathbf{x}_B^{#ij} \quad i, j \in \{0,1\} \end{aligned} \quad (10)$$

**Special cases.** In analogy to section 1.2, Equation 9 contains two special cases:

1. **Metabolic and Isotopic Stationary States:** Referring to Eq. 9 and assuming a metabolic and isotopic stationary  $d\mathbf{c}_{IM}^{Iso} = 0$  system state it follows:

$$\mathbf{0} = \mathbf{f}(\mathbf{v}^{\rightleftharpoons}, \mathbf{x}_{IM}^{Iso}, \mathbf{x}_S^{Iso}) \quad (11)$$

2. **Metabolic Stationary and Isotopic Non-stationary States:** Assuming only a metabolic stationary system state ( $dc_{IM}/dt \approx 0$ ) Equation 9 leads to:

$$\begin{aligned} c_{IM} \frac{d\mathbf{x}_{IM}^{Iso}}{dt} &= \mathbf{f}(\mathbf{v}^{\rightleftharpoons}, c_{IM}, \mathbf{x}_{IM}^{Iso}, \mathbf{x}_S^{Iso}), \\ \mathbf{x}_{IM}^{Iso}(0) &= \mathbf{x}_{IM,0}^{Iso} \end{aligned} \quad (12)$$

In both cases the function  $\mathbf{f}$  depends in a linear way on the fluxes  $\mathbf{v}^{\rightleftharpoons}$  and the substrate isotopomer fractions  $\mathbf{x}_S^{Iso}$  while it might be nonlinear in the intermediate isotopomer fractions  $\mathbf{x}_{IM}^{Iso}$ .

## 2 A Generalized Modeling Concept

### 2.1 Metabolic and Isotopic Non-stationary Systems

Here, we present a general modeling concept, which is based on Equation motivated by [16] in terms of isotopomers. In order to benefit from the increase in performance reported for the solution in  $^{13}\text{C}$ -MFA, we implemented the cumomer approach (cf. Figure 2):

$$\begin{aligned} \frac{d\mathbf{c}_{IM}^{Cum}}{dt} &= \mathbf{f}(\mathbf{v}^{\rightleftharpoons}, \mathbf{c}_{IM}^{Cum}, \mathbf{c}_S^{Cum}, \alpha), \\ \mathbf{c}_{IM}^{Cum}(0) &= \mathbf{c}_{IM,0}^{Cum} \end{aligned} \quad (13)$$

The function  $\mathbf{f}$  in Equation 13 is linearly dependent on the substrate cumomers  $\mathbf{c}_S^{Cum}$  and nonlinear with respect to the reaction rates  $\mathbf{v}^{\rightleftharpoons}(\mathbf{c}_{IM}^{Cum}, \alpha)$ . The use of cumomers instead of isotopomers is motivated by the fact that the nonlinear ODE system can be partitioned into cascaded subsystems of ODE's [18]. In short, all cumomers with equal weight (identical number of labeled C-atoms) form one level of system equations.

These levels are solved consecutively, starting from level 0, which is solely determined by the stoichiometry and corresponding kinetics (cf. Equation 7), up to the highest level, which is determined by the C-atoms of the longest carbon backbone occurring in the network.

To illustrate the concept consider the cumomer balances around the pool C of the running example (cf. Figure 2). For clarity, the elements of the cumomer concentration vector of C are denoted as

$$\mathbf{c}_C^{Cum} = (C\#xxxx, C\#xxx1, \dots, C\#1111)^T, x \in \{0,1\}$$

Corresponding cuomer fractions are introduced by  $\mathbf{x}_{IM}^{Cum} = \mathbf{c}_{IM}^{Cum} / c_{IM}$  and given in small letters, e.g.  $b\#x1$

level 0:

$$\frac{dC\#xxxx}{dt} = v_2^{\rightarrow} - v_3^{\rightarrow}$$

level 1:

$$\frac{dC\#xxx1}{dt} = v_2^{\rightarrow} \cdot e\#x1 - v_3^{\rightarrow} \cdot c\#xxx1$$

$$\frac{dC\#xx1x}{dt} = v_2^{\rightarrow} \cdot e\#1x - v_3^{\rightarrow} \cdot c\#xx1x$$

$$\frac{dC\#x1xx}{dt} = v_2^{\rightarrow} \cdot b\#x1 - v_3^{\rightarrow} \cdot c\#x1xx$$

$$\frac{dC\#1xxx}{dt} = v_2^{\rightarrow} \cdot b\#1x - v_3^{\rightarrow} \cdot c\#1xxx$$

level 2:

$$\frac{dC\#xx11}{dt} = v_2^{\rightarrow} \cdot e\#11 - v_3^{\rightarrow} \cdot c\#xx11$$

$$\frac{dC\#x1x1}{dt} = v_2^{\rightarrow} \cdot \boxed{b\#x1} \cdot \boxed{e\#x1} - v_3^{\rightarrow} \cdot c\#x1x1$$

$$\frac{dC\#x11x}{dt} = v_2^{\rightarrow} \cdot \boxed{b\#x1} \cdot \boxed{e\#1x} - v_3^{\rightarrow} \cdot c\#x11x \quad (14)$$

$$\frac{dC\#1xx1}{dt} = v_2^{\rightarrow} \cdot \boxed{b\#1x} \cdot \boxed{e\#x1} - v_3^{\rightarrow} \cdot c\#1xx1$$

$$\frac{dC\#11xx}{dt} = v_2^{\rightarrow} \cdot b\#11 - v_3^{\rightarrow} \cdot c\#11xx$$

level 3:

$$\frac{dC\#x111}{dt} = v_2^{\rightarrow} \cdot \boxed{b\#x1} \cdot \boxed{e\#11} - v_3^{\rightarrow} \cdot c\#x111$$

$$\frac{dC\#1x11}{dt} = v_2^{\rightarrow} \cdot \boxed{b\#1x} \cdot \boxed{e\#11} - v_3^{\rightarrow} \cdot c\#1x11$$

$$\frac{dC\#11x1}{dt} = v_2^{\rightarrow} \cdot \boxed{b\#11} \cdot \boxed{e\#x1} - v_3^{\rightarrow} \cdot c\#11x1$$

$$\frac{dC\#111x}{dt} = v_2^{\rightarrow} \cdot \boxed{b\#11} \cdot \boxed{e\#1x} - v_3^{\rightarrow} \cdot c\#111x$$

level 4:

$$\frac{dC\#1111}{dt} = v_2^{\rightarrow} \cdot \boxed{b\#11} \cdot \boxed{e\#11} - v_3^{\rightarrow} \cdot c\#1111$$

The cumomer fractions of level 0 ( $b\#xx = c\#xxxx = e\#xx = 1$ ) are not given explicitly. Considering the levels above 1 the framed cumomer fractions are already determined in lower levels. It should be noticed, that the information on the C-atom transitions, i.e. which C-atom of the substrate is transferred to which C-atom of the product, is essential for the correct formulation of the model equations.

Already, from this simple example it becomes clear, that the whole system equations can hardly be formulated manually. To avoid tedious and error prone typing it is desirable to generate systems like Equation 14 in a fully automatized way. We use the existing software toolbox 13CFLUX to take this task [19].

## 2.2 Kinetics for Labeling Dynamics

Basically for every type of mechanistic enzyme kinetics the separate formulation of steady-state forward and backward rates is possible applying e.g. the King-Altman method [6]. Although automatable, it is questionable if a description of one single reaction step by a complex mechanistic model like e.g. Bi-Bi Random Order comprising 18 rate constants to be fitted, results in an overall valid dynamic model.

For that reason we choose the convenience rate law [8] as a simplified mechanistic approach, which can be specified by a small number of parameters and is easy to handle for automatically assign a kinetic model to each reactions step of a dynamic network model. The general rate law of a convenience kinetic for  $n$  substrates and  $m$  products is given by [8]:

$$v = v_{max,1} \cdot \frac{\mathbf{v}_{max}^{\rightarrow} \cdot \prod_i^n \left( \frac{c_{S_i}}{K_{S_i}} \right) - \mathbf{v}_{max}^{\leftarrow} \cdot \prod_j^m \left( \frac{c_{P_j}}{K_{P_j}} \right)}{\prod_i^n \left( 1 + \frac{c_{S_i}}{K_{S_i}} \right) + \prod_j^m \left( \frac{c_{P_j}}{K_{P_j}} \right) - 1} \quad (15)$$

In case of a reaction where only one substrate is converted to one product the known Michaelis-Menten kinetic is derived (cf. Equation 8).

As an example, consider the cumomer mass balance around the pool B of the example (cf. Figure 2):

$$\begin{aligned} \frac{dc_{IM}^{Cum}}{dt} &= \mathbf{f}(\mathbf{v}^{\rightleftharpoons}, \mathbf{c}_{IM}^{Cum}, \mathbf{c}_S^{Cum}, \alpha), \\ c_{IM}^{Cum}(0) &= c_{IM,0}^{Cum} \\ \frac{dc_B^{#ij}}{dt} &= \frac{1}{1 + \frac{c_A}{K_{A,1}} + \frac{c_B}{K_{B,1}}} \cdot \\ &\cdot \left( \mathbf{v}_{max,1}^{\rightarrow} \cdot \frac{c_A}{K_{A,1}} \cdot \mathbf{x}_A^{#ij} - \mathbf{v}_{max,1}^{\leftarrow} \cdot \frac{c_B}{K_{B,1}} \cdot \mathbf{x}_B^{#ij} \right) + \\ &\quad + \frac{1}{1 + \frac{c_B}{K_{B,5}} + \frac{c_E}{K_{E,5}}} \cdot \\ &\cdot \left( \mathbf{v}_{max,5}^{\leftarrow} \cdot \frac{c_E}{K_{E,5}} \cdot \mathbf{x}_E^{#ij} - \mathbf{v}_{max,5}^{\rightarrow} \cdot \frac{c_B}{K_{B,5}} \cdot \mathbf{x}_B^{#ij} \right) - \\ &\quad - \mathbf{v}_{max,2}^{\rightarrow} \cdot \frac{c_B}{K_{B,2} + c_B} \cdot \mathbf{x}_B^{#ij}, i, j \in \{0,1\} \end{aligned} \quad (16)$$

For modeling metabolic regulation, generic terms for activation and inhibition can be multiplicatively combined with Equation 15:

$$\prod_a \left( \frac{k_{A_a} + c_{A_a}}{k_{A_a}} \right), \prod_b \left( \frac{k_{I_b}}{k_{I_b} + c_{I_b}} \right) \quad (17)$$

When parameterizing a dynamic model only the right combinations of kinetic parameter values and initial pool sizes will lead to a certain stationary state of the undisturbed system.

In order to use our modeling concept for the simulation of different experimental states (including metabolic stationarity) the dynamic network is coupled to a stationary network representing the initial system state.

Thereby the maximal reaction rates  $\mathbf{v}_{max}^{\rightleftharpoons} = \mathbf{f}(\mathbf{v}_0^{\rightleftharpoons}, \mathbf{c}_{IM,0}, \alpha)$  are formulated as functions for initial values of fluxes  $\mathbf{v}_0^{\rightleftharpoons}$  and intermediate concentrations  $\mathbf{c}_{IM,0}$  as well as kinetic parameters (now only comprising affinity constants).

Hence, the model includes one additional term per kinetic rate equation, e.g. for the forward step of reaction  $v_1$  the two equations are considered:

$$v_1^{\rightarrow} = v_{max,1}^{\rightarrow} \cdot \frac{\frac{c_A}{K_{A,1}}}{1 + \frac{c_A}{K_{A,1}} + \frac{c_B}{K_{B,1}}}, \quad (18)$$

$$v_{max,1}^{\rightarrow} = v_{1,0}^{\rightarrow} \cdot \frac{1 + \frac{c_{A,0}}{K_{A,1}} + \frac{c_{B,0}}{K_{B,1}}}{\frac{c_{A,0}}{K_{A,1}}},$$

### 3 An Universal Framework for Simulation and Evaluation of Metabolic Networks

So far we utilized a quite simple example network for explaining our modeling concept. Even for that system the number of cumomer mass balances and kinetic equations to be generated is quite high. A more realistic example network modeling reactions of glycolysis and pentose phosphate pathways already contains 682 ODE's for cumomer concentrations and a total of 112 kinetic parameters.

Clearly, manual generation of such models becomes infeasible and automated model code generation is strongly recommended. Additionally, the resulting dynamic model is a system of highly nonlinear ODE's and therefore demands for a simulation environment that can handle this complexity.

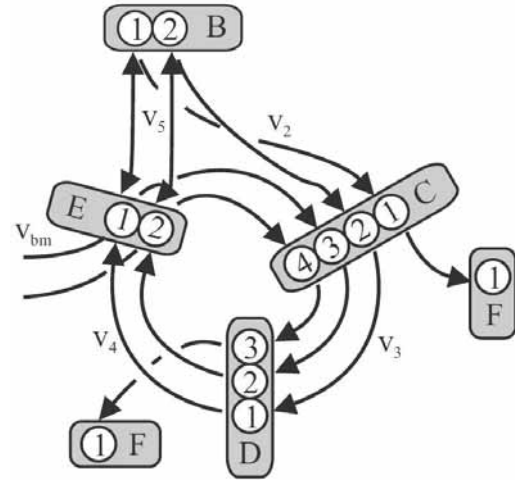
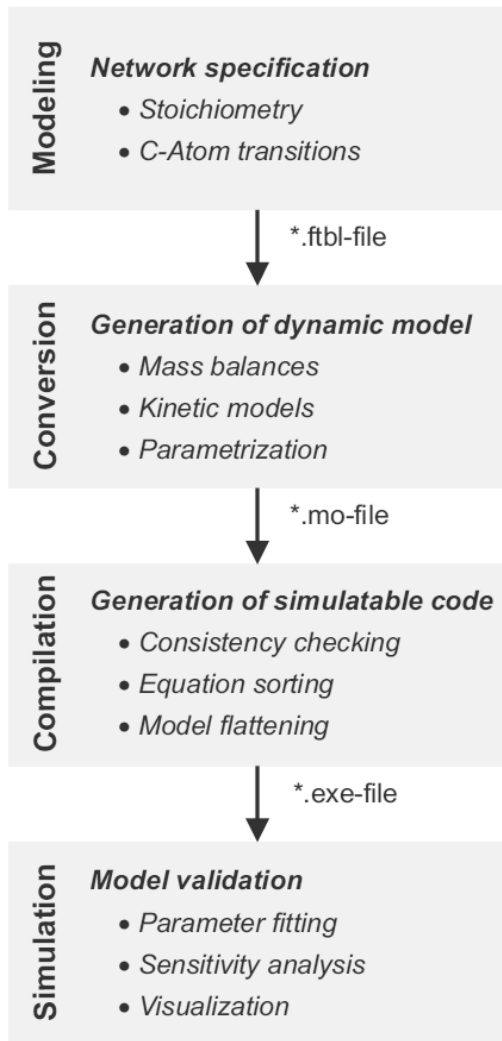
For that reason we developed a software framework, which allows the automatic generation of dynamic metabolic and isotopic network models of any size and complexity and, moreover, offers the perspective to use sophisticated methods for statistical analysis and interpretation of simulation results (cf. Figure 3).

As a core for setting up the model and building executable simulation code we use the Modelica language in combination with the Dymola environment (Dynasim AB, 5.0, [www.modelica.org](http://www.modelica.org)).

#### 3.1 Setup of Dynamic Model Equations

Starting from biochemical network descriptions (including stoichiometry, C-atom transitions, initial conditions) using the software 13CFLUX, consistent algebraic equation systems (AE) of mass balances for all cumomer pools are generated.

These AE's are then transformed into Modelica specific code consisting of ODE-systems for all cumomers, i.e. addition of a left hand side and kinetic equations (cf. Equation 13).



```

model dynamicModel
  //Declaration part
  //Model state
  parameter Integer flag=5;
  //Cumomer concentrations
  Real A_ix(start=0.005267);
  ...
  //Reaction rates
  Real v1_fwd;
  ...
  //Kinetic parameters
  parameter Real v1_km_A=1;
  ...
  equation
  //Balances
  der(A_x1)=vupt_fwd*S_x1/S_xx+...;
  ...
  //Kinetics
  v1_fwd=v1_fwd_vmax*(A_xx/v1_km_A)/...;
  ...
end dynamicModel;
  
```

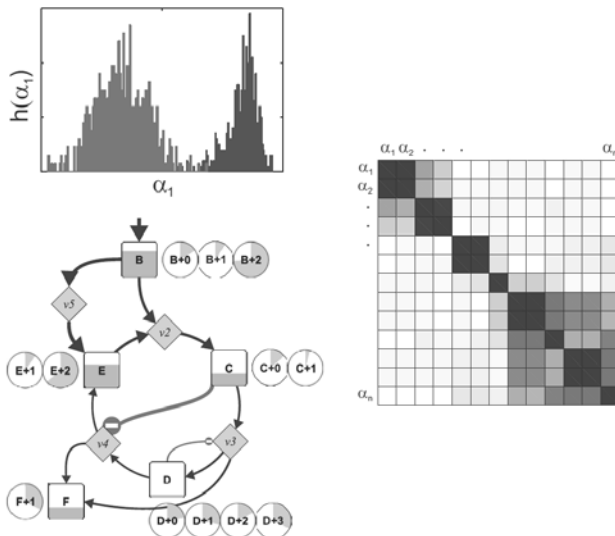


Figure 3: Workflow of model generation. Steps involved can be divided in four categories: (1) specification of model equations, (2) conversion to a Modelica file, (3) its compilation into an executable code which is then used for versatile simulation tasks (4).

**Definition of Model States.** Due to the general model structure (cf. Equation 13 and Section 2.3) in total five types of experimental states can be simulated using one model with different scenarios settings:

1. The external substrates  $\mathbf{c}_S^{Cum}$  are unlabeled and
  - do not change over time resulting in an AE-system ( $d\mathbf{x}_{IM}^{Cum}/dt = 0 = \mathbf{f}(\mathbf{v})$ ) depending only on steady state fluxes ( $\mathbf{v} = const$ ). This also equals the starting point, where the system is in a unlabeled metabolic stationary state ( $0 = \mathbf{f}(\mathbf{v}_0)$ ),
  - change over time resulting in an ODE-system ( $d\mathbf{x}_{IM}^{Cum}/dt = 0 \Rightarrow d\mathbf{c}_{IM}/dt = \mathbf{f}(\mathbf{v}, \mathbf{c}_{IM}, \mathbf{c}_S, \alpha)$ ) depending on dynamic intermediate concentrations ( $\mathbf{c}_{IM}$ ).
2. The external substrates substrates  $\mathbf{c}_S^{Cum}$  are based on specifically labeled mixtures  $\mathbf{x}_S^{Cum}$  and
  - do not change over time leading to an AE-system ( $d\mathbf{c}_{IM}^{Cum}/dt = 0 := \mathbf{f}(\mathbf{v}^{\equiv}, \mathbf{c}_{IM}^{Cum}, \mathbf{c}_S^{Cum}, \alpha)$ ) depending on steady state fluxes ( $\mathbf{v} = const$ ) and cumomer labeling fractions ( $\mathbf{x}_{IM}^{Cum} = const$ ),
  - the substrate cumomer fractions ( $\mathbf{x}_S^{Cum}$ ) change over time leading to an DAE-system
 
$$d\mathbf{c}_{IM}/dt = 0 \Rightarrow \mathbf{c}_{IM} \cdot d\mathbf{x}_{IM}^{Cum}/dt = \mathbf{f}(\mathbf{v}^{\equiv}, \mathbf{c}_{IM}, \mathbf{x}_{IM}^{Cum}, \mathbf{c}_S^{Cum})$$
 depending on steady state fluxes ( $\mathbf{v} = const$ ), dynamic cumomer labeling fractions ( $\mathbf{x}_{IM}^{Cum}$ ) and metabolite pool sizes ( $\mathbf{c}_{IM} = const.$ ),
  - change over time resulting in the general ODE-system of Equation 13.

**Manual Adaption of Model Code.** The initial parametrization of the automatically generated model code allows directly starting forward simulations of the different model states. In order to describe a real biochemical system under certain experimental conditions manual adaption of the following items are possible:

- **Kinetic types:** As the standard kinetic model for all reactions, the convenience rate law is chosen (cf. Section 1.2). Nevertheless, if other kinetic types for single reaction steps are needed, they can be easily substituted.

- **Substrate labeling mixture:** For each external substrate the standard mixture is formed by an amount of unlabeled, single labeled ( $1\text{-}^{13}\text{C}$ ) and fully labeled ( $\text{U-}^{13}\text{C}$ ) substrate. Different compositions can be fixed manually.

### 3.2 Model Validation

After generation, Modelica models can be directly compiled into highly efficient executable simulation code (cf. Figure 3). Model simulations, parameter fittings as well as comprehensive statistical evaluations are performed under MATLAB (Mathworks, R2008b) on a high-performance workstation under Linux.

**Sensitivity Analysis.** As an essential ingredient of model based inference, sensitivity analysis for model variables and parameters is performed using an automatic differentiation (AD) method developed for Modelica source code [3]. In short, ADModelica strives to semantically augment Modelica models with Modelica code for computing certain sensitivities, with minimal user efforts.

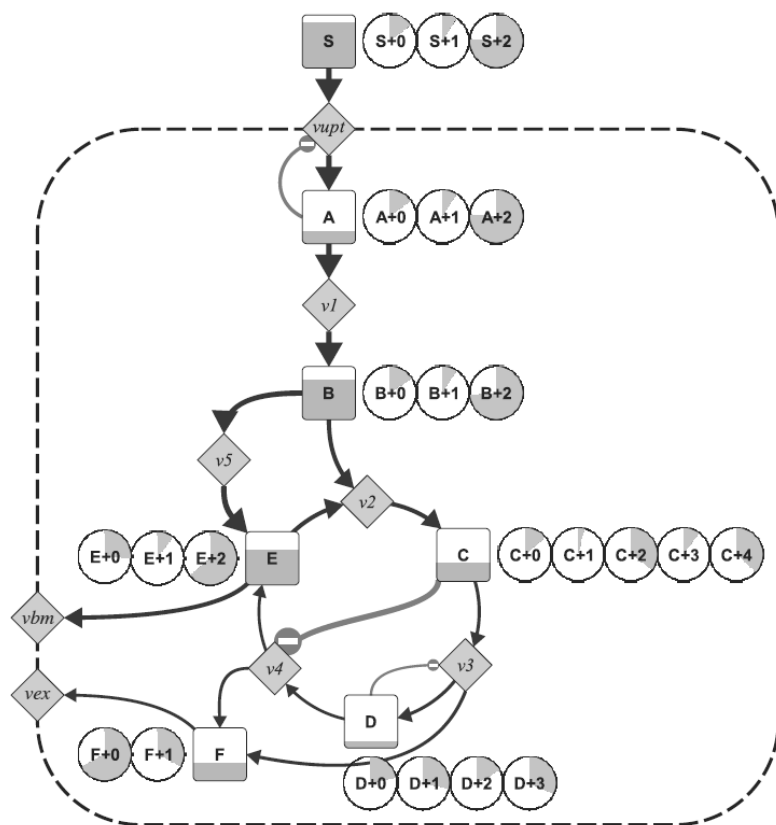
Aiming at the full-support of Modelica language constructs, the current version supports most basic constructs of Modelica. Clearly, the number of equations in a differentiated model ( $> 70000$  for a realistic model) increases proportionally with the number of model parameters.

**Visualization of Simulation Results.** For an intuitive interpretation of simulation data under dynamic as well as steady state conditions a network visualization approach can be applied.

Huge amounts of different kinds of simulation data, e.g. dynamically changing intermediate concentrations and labeling fractions can be analyzed in a quick and comprehensive way (Figure 4).

The visualization of regulatory interactions in a given metabolic network is based on a concept defining the Regulatory Strength of effectors regulating certain reaction steps [9].





**Figure 4.** Visualization of the example network showing a metabolic and isotopic snapshot during dynamic simulation. Scaled simulation data of intermediate concentrations and reaction rates are represented by correspondingly filled boxes and varying arrow widths, respectively. The labeling state is visualized in form of mass isotopomer fractions using pie charts. Regulatory interactions are visualized by red (inhibition) or green (activation, not included) circles and the corresponding strength of each effector is represented by the circle size.

## 4 Conclusions

The proposed software framework allows automatized generation of dynamic network models for simulating all currently realizable experiments in the fields of Metabolomics and Fluxomics, i.e. metabolic and isotopic (non-)stationary systems.

Due to the universality of the underlying modeling concept the different experimental states can be simulated using one software platform. Since, model setup and compilation is performed under the Modelica/Dymola environment realistic network models comprising more than 1000 ODE's can be simulated in acceptable time and therefore used for experimental validation.

The developed automatic differentiation method for Modelica code allows exact calculation of Jacobians that can be used for sensitivity analysis and gradient based optimization routines. Finally, the whole framework covers a visualization approach where the resulting simulation data can be easily interpreted in the network context.

## References

- [1] Antoniewicz, M. R., Kelleher, J. K., and Stephanopoulos, G. (2007). Elementary metabolite units (EMU): a novel framework for modeling isotopic distributions. *Metab Eng* 9, 1 (2007), 68–86. 1096-7176.
- [2] Degenring, D., Froemel, C., Dikta, G., and Takors, R. (2004). Sensitivity analysis for the reduction of complex metabolism models. *J Process Contr* 14, 7 (2004), 729–745.
- [3] Elsheik, A., Noack, S., and Wiechert, W. (2008). Sensitivity analysis of modelica applications via automatic differentiation. In *6th International Modelica Conference* (Bielefeld, Germany, 2008), p. 669.
- [4] Hadlich, F., Noack, S., and Wiechert, W. (2008). Translating biochemical network models between different kinetic formats. *Metab Eng* (2008). 1096-7184 (Electronic) Journal article.
- [5] Heijnen, J. J. (2005). Approximative kinetic formats used in metabolic network modeling. *Biotechnol Bioeng* 91, 534–45. 0006-3592.
- [6] King, E. L., and Altman, C. (1956). A schematic method of deriving the rate laws for enzyme-catalyzed reactions. *J Phys Chem* 60 (1956), 1375–1378.
- [7] Kresnowati, M. T., Suarez-Mendez, C. M., van Winden, W. A., Vangulik, W. M., and Heijnen, J. J. (2008). Quantitative physiological study of the fast dynamics in the intracellular pH of *Saccharomyces cerevisiae* in response to glucose and ethanol pulses. *Metab Eng* 10, 1 (2008), 39–54. 1096-7176 (Print) Journal Article Research Support, Non-U.S. Gov't.
- [8] Liebermeister, W., and Klippe, E. (2006). Bringing metabolic networks to life: convenience rate law and thermodynamic constraints. *Theor Biol Med Model* 3 (2006), 41. 1742-4682 (Electronic) Journal Article, Research Support, Non-U.S. Gov't.

- [9] Noack, S., Wahl, A., Qeli, E., and Wiechert, W. (2007). Visualizing regulatory interactions in metabolic networks. *BMC Biol* 5 (2007), 46. 1741-7007 (Electronic) Journal Article Research Support, Non-U.S. Gov't.
- [10] Nöh, K., Grönke, K., Luo, B., Takors, R., Oldiges, M., and Wiechert, W. (2007). Metabolic flux analysis at ultra short time scale: isotopically non-stationary <sup>13</sup>C labeling experiments. *J Biotechnol* 129, 2 (2007), 249–67. 0168-1656 (Print) Journal Article Research Support, Non-U.S. Gov't.
- [11] Oldiges, M., Kunze, M., Degenring, D., Sprenger, G. A., and Takors, R. (2004). Stimulation, monitoring, and analysis of pathway dynamics by metabolic profiling in the aromatic amino acid pathway. *Biotechnol Prog* 20, 6 (2004), 1623–33. 8756-7938 (Print) Evaluation Studies Journal Article Research Support, Non-U.S. Gov't.
- [12] Oldiges, M., Lütz, S., Pflug, S., Schroer, K., Stein, N., and Wiendahl, C. (2007). Metabolomics: current state and evolving methodologies and tools. *Appl Microbiol Biotechnol* 76, 3 (2007), 495–511, 0175-7598 (Print) Journal Article Review.
- [13] Schwender, J. (2008). Metabolic flux analysis as a tool in metabolic engineering of plants. *Curr Opin Biotechnol* 19, 2 (2008), 131–7. 0958-1669 (Print) Journal Article Research Support, U.S. Gov't, Non-P.H.S. Review.
- [14] Stephanopoulos, G., and Vallino, J. J. (1991). Network rigidity and metabolic engineering in metabolite overproduction. *Science* 252, 5013 (1991), 1675–81. 0036-8075 (Print) Journal Article Research Support, Non-U.S. Gov't Research Support, U.S. Gov't, Non-P.H.S. Review.
- [15] Van Winden, W. A., Heijnen, J. J., and Verheijen, P. J. (2002). Cumulative bondomers: a new concept in flux analysis from 2D [<sup>13</sup>C,<sup>1</sup>H] COSY NMR data. *Biotechnol Bioeng* 80, 7 (2002), 731–45. 0006-3592 (Print) Journal Article.
- [16] Wahl, S., Nöh, K., and Wiechert, W. (2008). <sup>13</sup>C labeling of experiments at metabolic nonstationary conditions: An explanatory study. *BMC Bioinf* 9, 152 (2008).
- [17] Wahl, S. A., Haunschild, M. D., Oldiges, M., and Wiechert, W. (2006). Unravelling the regulatory structure of biochemical networks using stimulus response experiments and large-scale model selection. *Syst Biol (Stevenage)* 153, 4 (2006), 275–85. 1741-2471 (Print) Journal Article.
- [18] Wiechert, W., Möllney, M., Isermann, N., Wurzel, M., and Degraaf, A. A. (1999). Bidirectional reaction steps in metabolic networks: III. Explicit solution and analysis of isotopomer labeling systems. *Biotechnology and Bioengineering* 66, 2 (1999), 69–85.
- [19] Wiechert, W., Möllney, M., Petersen, S., and Degraaf, A. A. (2001). A universal framework for <sup>13</sup>C metabolic flux analysis. *Metab Eng* 3, 3 (2001), 265–83. 1096-7176 (Print) Journal Article Research Support, Non-U.S. Gov't.

Submitted: March, 2010 – *SNE Accepted*

Submitted MATHMOD 2009: November 2008

Accepted MATHMOD 2009: January 2009

Submitted SNE: January 2010

Accepted: February 5, 2010

# Advanced Optimization of Hybrid-Electric-Vehicle Drivelines using Engine-in-the-Loop Simulation

Benjamin Tilch\*, Thorsten Reimers, Peter Eilts

Technische Univ. Braunschweig, Niedersächsisches Forschungszentrum Fahrzeugtechnik (NFF), Inst. of Internal Combustion Engines Braunschweig (ivb), Langer Kamp 6, 38106 Braunschweig, Germany; \*[b.tilch@tu-bs.de](mailto:b.tilch@tu-bs.de)

**Abstract.** At the Institute of Internal Combustion Engines of Technische Universität Braunschweig (ivb) a modular real-time longitudinal vehicle simulation model has been developed. The main goal of the vehicle model is the calculation of boundary conditions for the virtual development process of internal combustion engines (ICE). Especially in hybrid electric vehicle (HEV) development an efficient tool is necessary to solve the nearly unmanageable task to find the optimum configuration of hybrid topology, degree of hybridization and efficient operation strategy. To achieve optimal results several ICE models with different levels of detail, depending on the state of development, were used.

By coupling a real-time vehicle model and a hardware-in-the-loop (HiL) test bench engine characteristics like fuel consumption and exhaust gas emissions can be analysed. The use of engine-in-the-loop (EiL) provides the opportunity to integrate hard to simulate parameters in hybrid strategies. By using identical vehicle models for the power train design process and experimental studies at the test bench systematic errors can be avoided. Additionally the development time decreases. Furthermore the EiL method enables the development of any kind of hybrid power train and helps to design intelligent and unconventional hybrid strategies online.

## Introduction

Increasing oil prices, limited fossil resources and growing environmental requirements are a huge challenge in a world of rising demand for individual mobility. The aim of reducing fossil energy consumption and the greenhouse gas emission production demands alternative, innovative and efficient power train solutions. Beside the optimization of ICE's efficiency the combination with an electric motor in an HEV is a possibility to reach these targets.

Between conventional and electric vehicles almost all interim solutions are possible by combining ICE, EM and battery in hybrid structures as well as different degrees of hybridization. Figure 1 shows conceivable types of HEV power trains. To find the optimal combination in this parameter space an intelligent engineering method is obligatory. Because of the wide range of solutions a strategic method must be used to handle the almost unmanageable task of finding an optimal HEV configuration in minimum development time.

Based on the preliminary studies at the ivb [1], [2] a combined method of real-time simulation and experimental engine testing has been developed to manage this challenge.

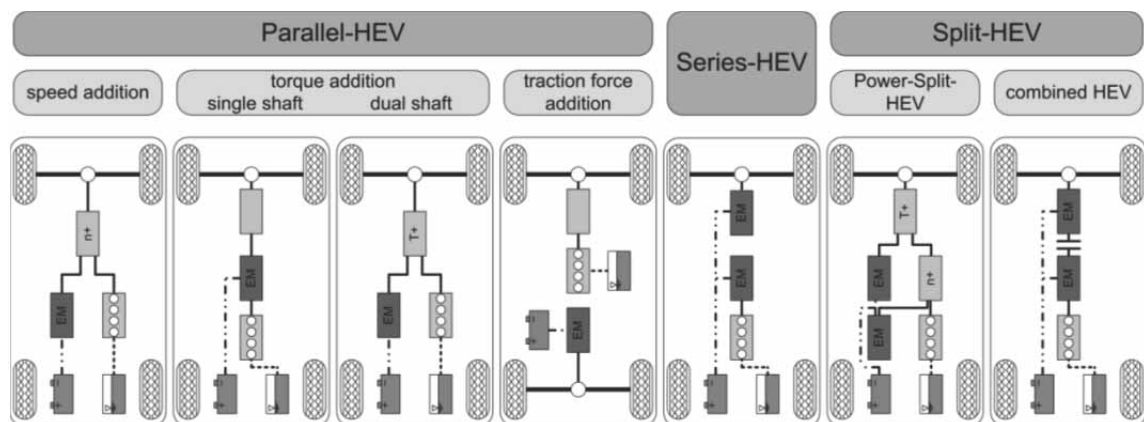


Figure 1. Overview of HEV power trains.

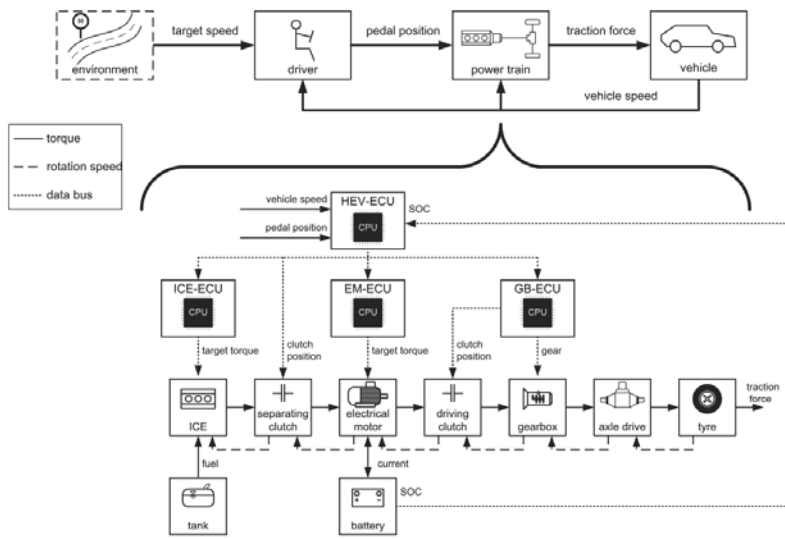


Figure 2. Configuration of the longitudinal HEV-model with parallel power train structure.

Besides different ICE models with different levels of detail a co-simulation with an engine test bench is possible. By replacing the simulated ICE by hardware at the EiL test bench research on the real engine behaviour for example considering thermal influences in engine friction or exhaust gas emissions can be done.

## 1 Vehicle Simulation Model and Engine-in-the-Loop test bench

### 1.1 Vehicle Simulation Model

For the experimental studies a longitudinal HEV drive train was modelled in Matlab/Simulink. An overview of the simulation model for a parallel HEV gives Figure 2.

The model is built up as a forward-looking-model and is divided in driver-, power train and vehicle model. In addition to the physical models of ICE, electrical motor (EM), battery, gearbox and axle drive, electronic control units (ECU) for ICE, EM and gearbox are necessary. Supervisory an HEV operating ECU manages the interaction of the other component-ECUs. The HEV-ECU splits the target torque between the ICE and the EM depending on the operation strategy. Furthermore the HEV-ECU triggers the two clutches between ICE and EM and between EM and gearbox.

The driver model acts as a PID controller. Due to comparison the current vehicle speed and the target speed the driver actuates the accelerator or brake pedal based on the control deviation.

In the power train model the traction force is calculated from the accelerator position. The power train model depends on the HEV structure and considers the losses, inertias and transmissions.

A tyre model transforms the brake pedal position into a brake force. In the vehicle model the available traction or brake force is balanced with the vehicle driving resistances.

In the model vehicle the longitudinal forces are balanced (Eq. 1). The resulting traction force depends on the driving resistances: rolling, air, gradient and acceleration resistant (Eqs. 2-6).

Traction force:

$$Z = \sum_{j=1}^n \frac{T_{tj}}{r_j} = F_{rol} + F_{air} + F_{gra} + F_{acc} \quad (1)$$

Rolling resistance:

$$F_{rol} = m_{veh} \cdot g \cdot f_r \quad (2)$$

$$f_r = f_{r,0} + f_{r,1} \cdot x_{veh} + f_{r,4} \cdot x_{veh}^4 \quad (3)$$

Air resistance:

$$F_{air} = c_w \cdot A \cdot \frac{\rho}{2} \cdot \dot{x}_{veh}^2 \quad (4)$$

Gradient resistance:

$$F_{gra} = m_{veh} \cdot g \cdot \sin \alpha \quad (5)$$

Acceleration resistance:

$$F_{acc} = \lambda \cdot m_{veh} \cdot \ddot{x}_{veh} \quad (6)$$

In consideration of the vehicle speed the driving power can be calculated (Eq. 7).

Driving power:

$$P_{veh} = Z \cdot \dot{x}_{veh} = (F_{rol} + F_{air} + F_{gra} + F_{acc}) \cdot x_{veh} \quad (7)$$

In the simulation model the traction force is totalled by the drive torques of ICE and EM divided by the dynamic tyre rolling radius. By converting Eq. 1 according to the unknown acceleration force and in consideration of the brake force  $F_{bra}$  from the tyre model the vehicle acceleration can be calculated (Eq. 8). The integration of the vehicle acceleration delivers the actual vehicle speed (Eq. 9). Integrating Eq. 9 results in driving distance (Eq. 10).

Vehicle acceleration:

$$\ddot{x}_{veh} = \frac{F_{acc}}{m_{veh}} = \frac{Z - F_{rot} - F_{air} - F_{gra} - F_{bra}}{m_{veh}} \quad (8)$$

Vehicle speed:  $\dot{x}_{veh} = \int \ddot{x}_{veh} dt \quad (9)$

Vehicle driving distance:  $x_{veh} = \int \dot{x}_{veh} dt \quad (10)$

Between ICE and EM a separating clutch is installed to realize electrical driving. A second clutch between EM and gearbox allows disconnection of ICE/EM and power train. This clutch allows battery charging during vehicle standstill.

The gearbox can be simulated as a simple or as a dual clutch transmission. In case of the dual clutch transmission the separate driving clutch is not required. The dual clutch transmission is divided physically in two partial transmissions with one clutch each.

Depending on the pedal position and the vehicle speed the gears of the partial transmissions are pre-chosen by the gearbox-ECU (GB-ECU). At a shifting event one clutch opens while the other clutch closes simultaneously. As a result the gear changes almost without traction interruption. The gearbox losses are interpolated from maps as a function of gear, speed and torque. In addition the torque flow direction is considered.

This is necessary to account for the difference between driven and non-driven (recuperation) operation mode. The EM efficiency is modelled by maps as a function of the operation point, which are divided in motor mode and generator mode. The input of the battery model is given by the supplied or removed EM electrical power. Additionally the charging and discharging efficiencies are considered. The stored energy of the battery is calculated by integration of the charging and discharging power. One of the most important inputs for the HEV-ECU for choosing an energy optimal operating mode is the battery state of charge (SOC). It is calculated from the quotient of actual energy content and nominal energy content.

Charging power:

$$P_{bat,charge} = \eta_{charge} \cdot P_{el,gen} \quad (11)$$

Discharging power:

$$P_{bat,charge} = \frac{1}{\eta_{discharge}} \cdot P_{el,mont} \quad (12)$$

Energy content:

$$E_{bat} = \int_{t_1}^{t_2} P_{bat} dt \quad (13)$$

State of charge:

$$SOC = \frac{E_{bat}}{E_{bat,nom}} \quad (14)$$

For the ICE three models with different levels of detail are integrated. Simple longitudinal design studies can be done with a map-based ICE model. In this simplest model the current ICE torque is calculated depending on the input target torque. To model the ICE transient behaviour, the inertia and the turbocharger response must be considered by correction functions which are calibrated by test bench measurements. The operation map limits for full load, friction curve, idle and maximum speed are considered by characteristic curves. Figure 3 shows characteristic operation maps of ICE and EM in a parallel HEV power train.

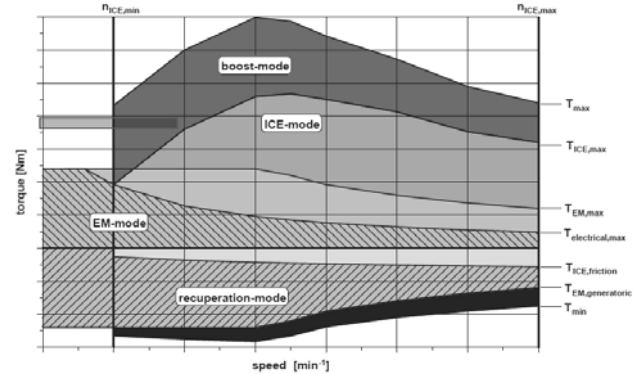


Figure 3. Operation maps of ICE and EM in a parallel HEV.

To simulate the transient thermal behaviour of the ICE, a thermal model can be used. The engine structure is build up as a lumped capacity model based on [3]. In this model the engine is divided in 18 capacities which interact via heat transfer network.

Furthermore a friction model has been developed which considers engine speed, coolant temperature, oil temperature and engine load depending on the ICE assemblies crankshaft, camshaft, piston and auxiliaries. The thermal losses of the combustion chamber are modelled as maps which are based on in-cylinder pressure analysis measured at an engine test bench. Simple models are used for the coolant and oil circuit.

For detailed analysis of the combustion process a complex thermodynamic model can be used which considers the gas dynamics in the inlet and outlet pipes. This model was set up with the software *GT Power* from *Gamma Technologies* and allows studying transient thermal effects of the combustion process like cooling down of the engine structure during stop phases.

For this purpose a simplified 3d-model of the engine structure allows detailed analysis of the thermal losses in the combustion chamber. In addition the model gives realistic information about the ICE fuel consumption. A detailed ICE-ECU is necessary for the combustion model to control fuel injection, turbo-charging and exhaust gas recirculation.

To verify the simulation results, the ICE can be replaced by real hardware at the EIL test bench. The residual vehicle model runs in real time on the automation system.

Parameters that are difficult to simulate like ICE exhaust gas emissions can be measured directly. This allows the evaluation of the exhaust gas emissions which depend on the operation mode considering the ICE efficiency. In this way the benefit of different operating modes can be rated under realistic conditions likewise. Figure 4 shows an overview of the ICE models. With proceeding availability different models like EM can be replaced by hardware at a power train test bench.

Superior to all components with specific ECUs the HEV-ECU manages the vehicle operation mode and the corresponding torque split between ICE and EM. The HEV-ECU first identifies the required driving power due to the vehicle speed and pedal positions. Based on this input parameters the HEV-ECU chooses from following operating modes (Fig. 3):

- **Electrical driving:** The EM delivers the necessary driving power while the ICE is shut off and the separating clutch is open. Fig. 3 clarifies that not the complete available EM power is useable for electrical driving. A safety range is necessary to get over the friction and inertia losses at the next ICE start procedure.
- **ICE start:** If the required driving power exceeds the electrical driving limit, the ICE has to start. The EM speeds up the ICE to actual power train speed.
- **Boost mode:** In boost mode the EM assists the ICE at transient conditions. For example the boost mode can improve the vehicle acceleration or smooth the transient ICE load. For this mode a high SOC is necessary.

- **ICE Load Increasing:** At low SOC the EM can be operated in generator mode to charge the battery. In this case the ICE load exceeds the required torque. This implies that the ICE works in a better specific fuel consumption operating point (Fig. 5). This operating mode requires a low SOC to avoid exceeding maximum battery energy content.
- **Recuperation:** In deceleration phases the EM can recuperate electrical energy from the kinetic energy of the vehicle. In this case the separating clutch opens to minimize ICE friction losses. Simultaneously the ICE is shut off.
- **ICE single operation mode:** This mode is equal to conventional vehicles. The prerequisite for this mode is that the required power exceeds maximum electrical driving power and SOC limitation so that no ICE load increasing is necessary.
- **Battery loading at standstill:** If the SOC drops below a critical limit the ICE can charge the battery at standstill phases. The separating clutch is closed while the driving clutch is open. The EM acts as a generator.

In addition special operating modes like *smoothing transient engine load* or *limiting the maximum ICE torque*, which are presented in [2], can be implied easily due to the modular model structure.

Figure 5 shows different operating strategies depending on the HEV power train structure compared to a conventional vehicle. The maps show the operating areas during New European Driving Cycle (NEDC). In a parallel HEV structure only load shifting is possible. A power split HEV allows torque and speed shifting of ICE operating point.

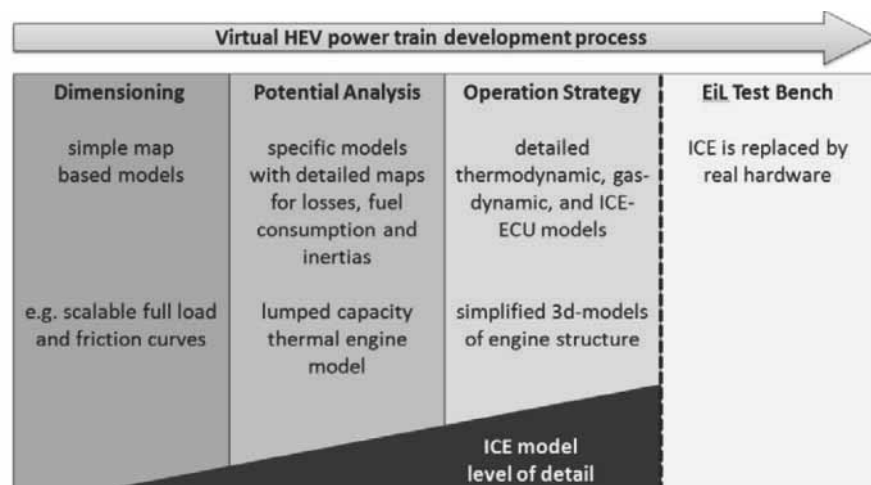


Figure 4. Available ICE models with different level of detail.

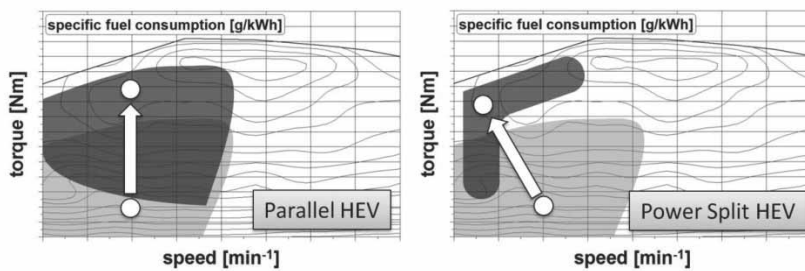


Figure 5. Operating point shifting depending on the HEV power train structure.

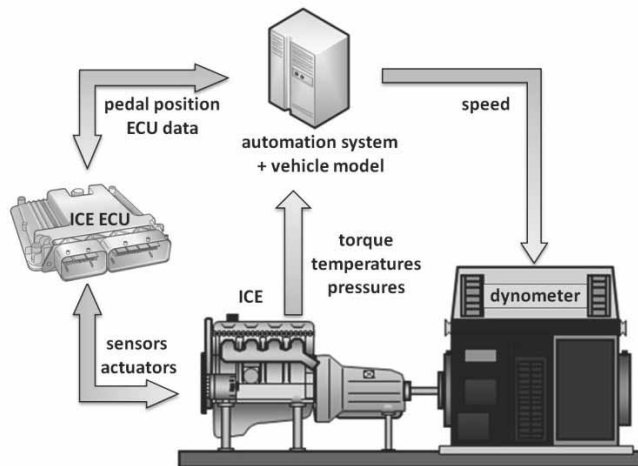


Figure 6. Engine-in-the-Loop test bench.

## 1.2 Engine-in-the-Loop Test Bench

At the engine test bench the virtual ICE is replaced by real hardware. The remaining submodels like EM, battery or gearbox are simulated in real time directly at the test bench control PC. Compared to other co-simulation real-time systems, this solution generates the shortest cycle times. This procedure ensures that the real-time models which are used in the experimental setup are identical to the models in the early virtual development process.

For the experimental studies a turbocharged gasoline engine is used. The ICE is coupled to a dynamometer which is a highly dynamic asynchronous load machine with a maximum speed of 10.000 rpm and a maximum torque of 450 Nm. Figure 6 gives an overview of the test bench configuration. The automation system controls the dynamometer and auxiliary systems like air and fluid conditioning. A test bench controller regulates both the dynamometer speed and the accelerator pedal position. Target values are calculated by the simulation model. The test bench sensors measure the ICE torque which is the input for the model to calculate the vehicle

acceleration. ECU variables can be changed online by the simulation model after coupling the ICE control unit with the automation system. The ICE control unit communicates with the ICE by sensors and actuators. To consider ICE operation values like temperatures and pressures in HEV operation strategy the measured data are available in the test system. All measured and simulated data are recorded with up to 1 kHz.

## 2 Simulation results

Figure 7 shows test bench results of a co-simulation of a parallel HEV power train. In this example the ICE data are measured on real hardware while the electrical components and a dual clutch transmission are simulated. The figure shows the extra-urban part of the NEDC. In the first seconds the vehicle accelerates from standstill to a constant vehicle speed of 70 kph. By exceeding the electrical power limitation the ICE begins to operate.

To reach the target speed the GB-ECU shifts up to sixth gear. After reaching the target speed the vehicle is driven by the ICE. The GB-ECU now shifts up because of a decrease in the accelerator pedal position. Then the target speed decreases to 50 kph. Because of the constant power requirement and high SOC the ICE is shut off. The electrical power is sufficient for electrical driving. An obvious decrease of SOC is observable. In the following acceleration the power demand exceeds the limit of electrical driving which necessitates the ICE's restart. While the ICE starts a peak in EM torque can be seen. The high EM torque results from the EM generating driving power and power for the ICE start procedure at the same time.

The following acceleration phase is passed in ICE mode until the HEV-ECU shifts the operation mode to *increasing ICE load* in order to charge the battery. Compared to the first constant speed phase of 70 kph, where no *ICE load increase* is used, the SOC decreased below the limit of 50%. Beyond this limitation the operation mode *ICE load increase* is enabled. To avoid a direct engine stop after exceeding this limitation a hysteresis is considered.

During *ICE load increasing* the EM operates in generator mode to reach a higher ICE load compared to the required torque. This generates a benefit in specific ICE efficiency and a simultaneously increasing SOC. At the end of the driving cycle the vehicle decelerates from 120 kph to standstill. While deceleration the EM works in generator mode and recuperates electrical energy from kinetic energy. To maximize the storable energy the ICE is shut off and the separating clutch opens to avoid ICE friction losses. This lifts up generator torque at the EM.

### 3 Conclusion

At the ivb a real-time HEV power train simulation model has been developed and adapted for co-simulation use at an EiL test bench. Depending on the development process models of different levels of detail are used. The EiL test bench offers the possibility that HEVs with different structures, degrees of hybridization and operation strategies can be analysed under realistic boundary conditions. This way parameters like exhaust gas emissions can be considered, that are difficult to simulate. The HEV-ECU can use both ICE parameters to optimize the operating strategy with focus on minimum fuel consumption and can adapt ICE-ECU parameters online. In future work the development method can be used as an efficient tool to evaluate different HEV power train solutions with an ICE within shortest time.

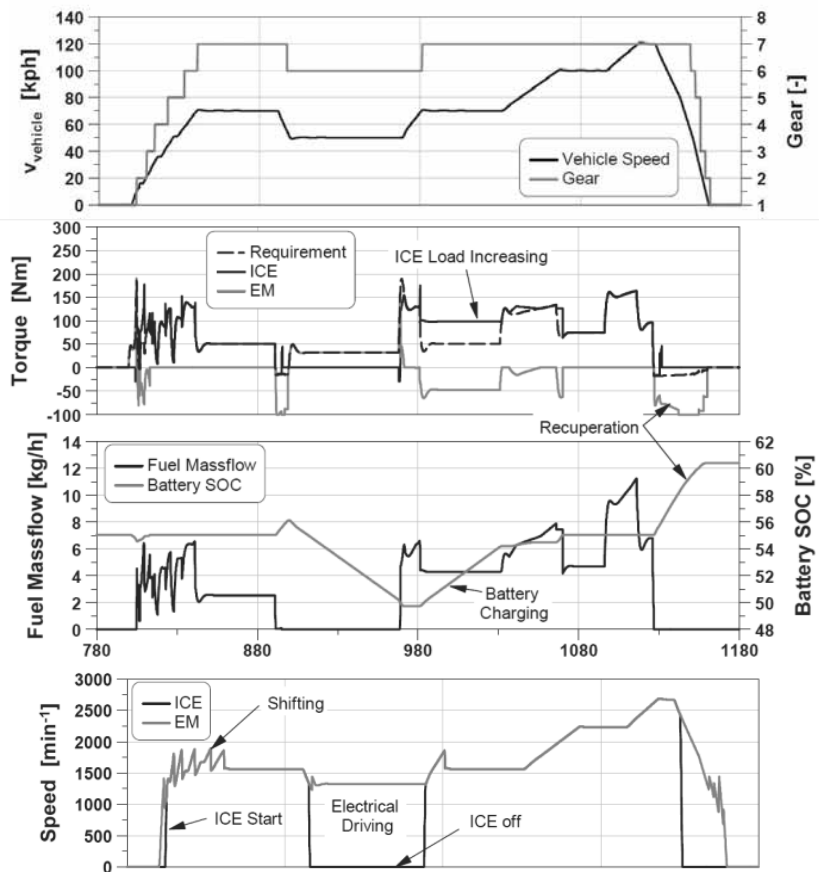


Figure 7. Example of a co-simulated test bench measurement.

A	vehicle frontal area (m <sup>2</sup> )	SOC	state of charge (%)
c <sub>w</sub>	drag coefficient (-)	T	torque (Nm)
E <sub>bat</sub>	energy content (Wh)	x <sub>veh</sub>	vehicle distance (m)
F <sub>acc</sub>	acceleration resistance (N)	ẋ <sub>veh</sub>	vehicle speed (m/s)
F <sub>air</sub>	air resistance (N)	ẍ <sub>veh</sub>	vehicle acceleration (m/s <sup>2</sup> )
F <sub>bra</sub>	braking force (N)	Z	traction force (N)
F <sub>gra</sub>	gradient resistance (N)		
f <sub>r</sub>	rolling resistant coefficient (-)	α	climbing angle (°)
F <sub>rol</sub>	rolling resistance (N)	η	efficiency (%)
g	acceleration due to gravity (m/s <sup>2</sup> )	λ	rotating mass coefficient (-)
m <sub>veh</sub>	vehicle mass (kg)	ρ	air density (kg/m <sup>3</sup> )
P <sub>bat</sub>	battery power (W)		
P <sub>veh</sub>	road resistance power (W)		

Nomenclature

### References

- [1] Lindenkamp, N. (2011). Strategien zur Reduzierung der NOX- und Partikelemissionen eines Dieselhybridfahrzeugs Dissertation, Technische Universität Braunschweig, Sierke Verlag, 2011
- [2] Lindenkamp, N.; Stöber-Schmidt, C.-P. and Eilts, P. (2009). Strategies for Reducing NOX- and Particulate Matter Emissions in Diesel Hybrid Electric Vehicles. *SAE-paper* 2009-01-1305.
- [3] Samhaber, C. (2002). *Simulation des thermischen Verhaltens von Verbrennungsmotoren*. Dissertation, Technische Universität Graz.

Submitted: September 2011 (ASIM STS Winterthur)

Accepted: July 20, 2012



# Software for Higher-order Sensitivity Analysis of Parametric DAEs

Moritz Schmitz<sup>1\*</sup>, Ralf Hannemann-Tams<sup>1</sup>, Boris Gendler<sup>2</sup>, Michael Förster<sup>2</sup>,  
Wolfgang Marquardt<sup>1</sup>, Uwe Naumann<sup>2</sup>

<sup>1</sup> Aachener Verfahrenstechnik, Process Systems Engineering, RWTH Aachen University,  
Templergraben 56, 52056 Aachen, Germany; \*moritz.schmitz@avt.rwth-aachen.de

<sup>2</sup> LuFG Informatik 12: Software and Tools for Computational Engineering, RWTH Aachen University, Germany

**Abstract.** We introduce AC-SAMMM (The Aachen platform for Structured Automatic Manipulation of Mathematical Models), a new software infrastructure for efficient transformation and evaluation of expressions and their higher-order derivatives. We describe the way this software can be used to perform automatically the translation of a model written in an equation-oriented language like Modelica into a subset of C/C++ and the generation of the model's higher-order derivative code by algorithmic differentiation (AD) techniques. The derivatives are generated, using the *derivative code compiler (dcc)*, an AD tool which provides source code transformation for a restricted but numerically relevant subset of C/C++. *dcc* can be applied repeatedly to its own output, to generate derivative codes of arbitrary order.

## Introduction

Many engineering applications exhibit the need of modeling and simulating increasingly complex problems. Therefore, many high-level equation-based modeling languages like Modelica ([www.modelica.org](http://www.modelica.org)), gPROMS ([www.pscenterprise.com/gproms](http://www.pscenterprise.com/gproms)) and SBML ([sb ml. org](http://sbml.org)) have been developed to formulate mathematical models in an intuitive, equation-based representation that enhance model development and maintenance. The usage of these models in different kinds of optimization problems becomes increasingly important.

These problems include model parameter estimation, optimal design of experiments for parameter estimation or model structure discrimination, design optimization, model-predictive control and real-time optimization. The solution of each of these problems typically requires the evaluation of symbolic expressions involving different type and order of derivatives [6], that are in general not provided by the modeling tools.

To bridge this gap, the software package AC-SAMMM (The Aachen platform for Structured Automatic Manipulation of Mathematical Models) has been jointly developed by AVT.PT (*Process Systems Engineering*) and STCE (*Software and Tools for Computational Engineering*) at RWTH Aachen University.

The aim of this project is to provide a software platform that generates robust, well-documented, and highly optimized model and derivative code by automatic manipulation of mathematical models independently of the model representation language used.

## 1 AC-SAMMM Introduction

Applying the AC-SAMMM platform, by means of algorithmic differentiation [5, 9], the user is able to generate higher-order derivatives of parametric semi-explicit index-1 differential-algebraic equations,

$$\begin{aligned}\dot{x} &= f(x(t), y(t), u(p, t)), \\ 0 &= g(x(t), y(t), u(p, t)),\end{aligned}\tag{1}$$

in an automatic manner. Here,  $t$  denotes the independent (time) variable,  $x(t) \in \mathbb{R}^{n_x}$  the differential or state variables,  $y(t) \in \mathbb{R}^{n_y}$  the algebraic variables and  $u(t, p) \in \mathbb{R}^{n_u}$  the inputs which are parameterized by some parameters  $p \in \mathbb{R}^{n_p}$  for convenience and without loss of generality.

Additional drivers are provided that support the efficient evaluation of the generated derivatives and the original mathematical model. The generated code is compiled into a dynamic link library or a shared object that can be accessed by third-party software (e.g. optimization and simulation software) via an C++ interface.

As illustration of the importance of fast and accurate higher-order derivatives, we consider *optimal control problems* and their solution by means of control vector parameterization (cf. Bruschi [2]).

In optimal control, the user has to control a dynamic process such that an objective function is minimized or maximized and usually some process constraints have to be satisfied. We consider the following class of Mayer-type optimal control problems (**OCP**), already discretized by means of control vector parameterization, resulting in the finite-dimensional nonlinear program (NLP).

$$\min_{p, t_f} \Phi(x(t_f), y(t_f)) \quad (2)$$

$$s. t. \quad \dot{x} = f(x, y, u(p, t)), \quad t \in [t_0, t_f] \quad (3)$$

$$0 = g(x(t), y(t), u(p, t)) \quad (4)$$

$$0 = x(0, p) - x_0(p) \quad (5)$$

$$0 \geq h(x(t_k), y(t_k), u(p, t_k)), k = 1, \dots, N \quad (6)$$

$$0 \geq e(x(t_f), y(t_f), u(p, t_f)) \quad (7)$$

This NLP is typically solved by gradient-based optimization methods such as SQP or interior point solvers. In general, these solvers need at least first-order derivatives of the objective function and the constraints and sometimes also second-order derivatives to form the corresponding Hessian of the Lagrangian. Therefore, an efficient and accurate calculation of these derivatives is fundamental when it comes to fast and accurate numerical calculations.

The paper is organized as follows. First, the typical workflow of AC-SAMMM is presented. Then, we explain the design of the `dcc`-generated derivative codes and show how they are called in an efficient way. In Section 4, details about the workflow and the functionality of the ESO are presented. Section 5 discusses an application of AC-SAMMM to a small OCP. Section 6 concludes the paper and gives directions for future development.

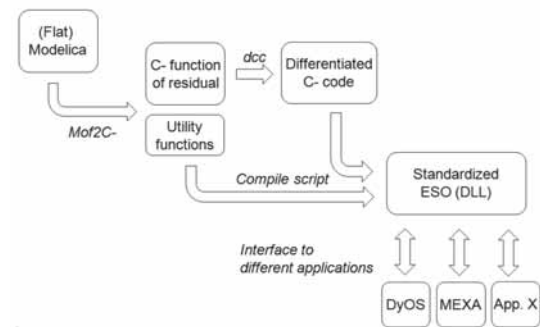
## 2 Typical Workflow in AC-SAMMM

Today, AC-SAMMM is able to translate mathematical models written in flat Modelica code into C- code using `Mof2C-` (a parser developed at AVT.PT) and to generate derivative code of arbitrary order by applying the AD-tool `dcc` [12] (developed at STCE) to the generated code.

Flat Modelica prescribes code where all object-oriented and hierarchical features of Modelica are eliminated ('flattened'). Though not really standardized in the language specification, the flattening process is considered to be part of all executable Modelica simulation environments (see [8, Chapter5]). C- is a small subset of the programming language C/C++. Additional algebraic manipulations will be applied and a highly optimized dynamic library will be obtained.

After the application of AC-SAMMM the original model information and additional derivative information (restricted to second- order derivatives today) are stored in the dynamic library. The access to this information is managed by a so-called *Equation Set Object* (ESO) which is an instance of a standardized C++ class.

The AC-SAMMM ESO is slightly modified variant of the ESO specified by the CAPE-OPEN organization in [11]. This ESO encapsulates all calls to the model residuals and related derivative functions and can be interfaced to different applications. The typical workflow of generating such an ESO is shown in Figure 1.



**Figure 1.** Workflow of AC-SAMMM. The flat Modelica code is translated to C- by `Mof2C-`. The C- files contain the residual function and some utility functions providing amongst others the variable and parameter names and the initial values for the variables (if algebraic variables are existent, their initial values are not necessarily consistent with the algebraic equations). In a second step, the residual function is differentiated by `dcc`. Finally all generated functions are compiled into a dynamic library defining the ESO-functions. The ESO defines the interface to third-party applications.

## 3 The Derivative Code Compiler `dcc`

For the reasons stated in Section 1, most optimization tools need higher-order derivatives of the dynamic model. AD source code transformation provides derivative code of arbitrary order. The derivative values computed by this code are accurate up to machine accuracy as opposed to the approximated values computed by finite differences.

Derivatives can be generated in the tangent-linear or in the adjoint mode. We define  $z = (x, y)$  and consider

$$F(z, \dot{z}, u) = \begin{pmatrix} \dot{x} - f(x, y, u) \\ g(x, y, u) \end{pmatrix} = 0$$

The software package AC-SAMMM uses the derivative code compiler `dcc` to generate the derivative code of an input code in tangent-linear or in adjoint mode. The input code is given in a subset of the programming language C/C++ called C-. `dcc` is able to use its own output again as input what allows the generation of derivative code of arbitrary order. The first-order tangent-linear model is defined as

$$\mathbb{R}^{n_x+n_y} \ni F^{(1)} \equiv F'(z, \dot{z}, u) \begin{pmatrix} z^{(1)} \\ \dot{z}^{(1)} \\ u^{(1)} \end{pmatrix} \quad (8)$$

where  $(z^{(1)}, \dot{z}^{(1)}, u^{(1)})^T \in \mathbb{R}^{2(n_x+n_y)+n_u}$

Given the implementation of  $F$  in C- `dcc` generates code that takes  $(z^{(1)}, \dot{z}^{(1)}, u^{(1)})^T$  as input and computes  $F^{(1)}$  with computational costs of about twice as high as the computational costs of  $F$ . The corresponding C++ implementation that is part of the ESO (see Section 4) has the following signature, given in Listing 1, where the prefix `t_1` stands for *first-order tangent-linear* and has the same meaning as the superscript <sup>(1)</sup>.

```
void eval_t_1 residuals ( double* t1_z,
double* t1_der_z, double* t1_der_u,
double* t1_residuals )
```

Listing 1. ESO-function calling first-order tangent-linear model.

The output of the function representing  $F^{(1)}$  is `t1_residuals`. All other arguments are the user's input. The actual values of the states and the parameters are members of the class providing this implementation and are known implicitly by the method.

The first-order adjoint model's output

$(z_{(1)}, \dot{z}_{(1)}, u_{(1)})^T \in \mathbb{R}^{2(n_x+n_y)+n_u}$  is defined as

$$\begin{pmatrix} z^{(1)} \\ \dot{z}^{(1)} \\ u^{(1)} \end{pmatrix} \equiv \begin{pmatrix} z^{(1)} \\ \dot{z}^{(1)} \\ u^{(1)} \end{pmatrix} + (F'(z, \dot{z}, u))^T F_{(1)} \quad (9)$$

where the vector  $(z_{(1)}, \dot{z}_{(1)}, u_{(1)})^T$  has to be given as input and will be overwritten by the new output value. Note that we use parenthesized superscripts to denote tangent-linear projections and parenthesized subscripts to mark adjoint projections. The ESO-implementation is analogous to that of the tangent-linear model with prefix `a_1` corresponding to subscript <sub>(1)</sub>.

The computational costs of the adjoint model are about six times higher than the costs for evaluating  $F$  (for an estimation of the computational costs, see [5]).

Nevertheless, in case of a matrix vector product where the matrix is the transposed Jacobian, the adjoint model is the best choice. The only alternative to compute this matrix vector product would be the accumulation of the whole Jacobian to be able to transpose it afterward.

The second-order tangent over adjoint model's output  $(z_{(1)}^{(2)}, \dot{z}_{(1)}^{(2)}, u_{(1)}^{(2)})^T \in \mathbb{R}^{2(n_x+n_y)+n_u}$  is defined as

$$\begin{pmatrix} z_{(1)}^{(2)} \\ \dot{z}_{(1)}^{(2)} \\ u_{(1)}^{(2)} \end{pmatrix} \equiv \begin{pmatrix} z_{(1)}^{(2)} \\ \dot{z}_{(1)}^{(2)} \\ u_{(1)}^{(2)} \end{pmatrix} + \langle F_{(1)}, F''(z, \dot{z}, u) \begin{pmatrix} z^{(2)} \\ \dot{z}^{(2)} \\ u^{(2)} \end{pmatrix} \rangle + F'(z, \dot{z}, u)^T F_{(1)}^{(2)}$$

where the second term is the projection of the  $((n_x + n_y) \times (2(n_x + n_y) + n_u) \times (2(n_x + n_y) + n_u))$  Hessian tensor  $F''(z, \dot{z}, u)$  in directions  $F_{(1)} \in \mathbb{R}^{n_x+n_y}$  and where  $(z_{(1)}, \dot{z}_{(1)}, u_{(1)})^T \in \mathbb{R}^{2(n_x+n_y)+n_u}$ . The ESO-method that accesses the `dcc`-generated second-order model is shown in Listing 2. To exploit structural facts of the Jacobian and Hessian we use coloring algorithms to find possibilities to compress the Jacobian and Hessian [14, 15].

```
void eval_t_2_a1 residuals ( double* t2_z,
double* a1_z, ..., double* a1_f,
double* t2_a1_f )
```

Listing 2. ESO-function calling second-order tangent over adjoint model.

## 4 AC-SAMMM in Practice

The above mentioned Jacobian compression techniques are used in the AC-SAMMM Jacobian and Hessian driver routines and are automatically executed if the user calls one of the ESO-methods to access derivatives. For example, the ESO-method

```
void get_jacobian_values ( long len ,
long* indices, double* jacobian ) ;
```

calls the first-order tangent-linear code (see assignment 8) with optimal seeding vectors  $(z_{(1)}, \dot{z}_{(1)}, u_{(1)})^T$  to get the specified Jacobian entries with minimal computational effort. The argument `len` defines the length of the output array *jacobian*.

The entries in *indices* define the relative indices of the wanted non-zero elements in the jacobian matrix arranged in row-major format  $\left(\frac{\partial F}{\partial z} \frac{\partial F}{\partial u}\right)$ .

The ESO defines the C++ interface to all client-applications using AC-SAMMM. The AC-SAMMM-ESO definition is based on the CAPE-OPEN ESO interface and contains some additional functions. The ESO is intended to cover at least the required functions to solve a differential-algebraic system and additional functions to handle higher-order derivatives.

So far AC-SAMMM can only be used on Microsoft Windows platforms, but AC-SAMMM will be adapted to deal with multiple platforms such as *Microsoft Windows*, *Linux/Unix* and *Mac OS*, soon. In order to accomplish this, AC-SAMMM is developed using CMake [7]. The way starting from a dynamic model coded in Modelica to the dynamic library including the higher-order derivatives and the original model information is completely automated (see Figure 1). The user calls a script that takes as argument the name of the flat-Modelica model. The flat-Modelica parser `MOE2C-` creates the C-code of the model. Beside the residual function, that is presented in detail in Section 5, other important files are generated, amongst others the *block residual function*.

The block residual function provides a signature with an index-set to define the indices of the residuals to evaluate. This offers an important gain in computational time if only a small subset of the residual vector should be evaluated. This is the case if, for example, a block-decomposition routine for the determination of consistent initial values is used. Additionally, `dcc` uses operator overloading techniques in combination with the propagation of bit patterns to determine the sparsity pattern of the Jacobian.

The next fully automated step during the script call is the call of `dcc`. The first- and second-order derivatives of the residual-model are generated and will be used later on to define the ESO-functions (see Listing 1 and Listing 2). AC-SAMMM is restricted to second-order derivatives (tangent over adjoint mode) but could be easily modified to provide even higher-order derivatives. In the final step, the dynamic library is created using a C++ compiler (which has to be installed on the user's PC). The hereby generated model library can then be loaded dynamically (see Listing 4) into the AC-SAMMM-ESO providing amongst others the efficient drivers for derivative evaluation up to second order by exploiting sparsity.

Some examples of ESO-methods have been presented so far. For a complete description on all ESO-methods we refer to the AC-SAMMM manual [13].

## 5 Case Study

In the introduction we mentioned the problem class *optimal control problems*. In this section we deal with a problem belonging to this class that will serve as a first proof of concept.

We consider the illustrative and very simple problem of a car that has to travel a fixed distance in minimal time. In addition, the car has to start at rest and its final velocity has to be zero. The control parameter is its acceleration  $u(t)$  and with parameter  $\alpha$

$$\begin{aligned} \min_{u(t)} t_f \\ \dot{x} = v \quad x(t_f) > x_{t_f} \\ \dot{v} = u - \alpha v^2 \quad v(t) \leq v_{max} \\ x(0), v(0) = 0 \quad -1 \leq u(t) \leq 1 \\ v(t_f) = 0 \end{aligned}$$

One can easily verify that this problem formulation fits the generic form **OCP**, if an additional state representing the time is introduced. The corresponding (flat) Modelica code for the dynamic model is given in Listing 3.

```
model car
parameter Real accel = 2;
parameter Real alpha = 0.0025;
Real ttime; Real velo; Real dist;
equation
der(dist) = velo;
der(velo) = accel - alpha *pow(velo, 2);
der(ttime) = 1;
end car;
```

Listing 3. Flat Modelica code for case study.

The Modelica model represents only the dynamic model that defines neither an objective function nor additional constraints, so the control  $accel = u(t)$  is defined as a constant parameter. The core-capacity of AC-SAMMM is the manipulation of dynamic models of the form (1). Hence, we will first explain the treatment of manipulating the dynamics before dealing with the entire optimization problem.

The script call that executes the operations mentioned in Section 4 has to be entered in a command-line interface being able to interpret BASH-commands (i.e. cygwin ([www.cygwin.com](http://www.cygwin.com)) for Windows platforms). If the model's name is *car.mof*, then the script is being called by the command `acsammm car`. The intermediate C- functions generated by `MOE2C-` contain all the model information provided by the Modelica model. The function representing the residual function (shown in Listing 4) will be processed by `dcc`.

```

void res (double * yy, double * der x,
         double * x, double * , int &n_x, int &n_p)
// $ad indep x p; .....// $ad dep yy
{ // .....// scalar equation 1
  acs pow(var_velo, 2, var pow0);
  yy[i_E] = der_var_velo -
    (par_accel - par_alpha * var_pow0);
  i_E = i_E + 1; // / ..... }

```

Listing 4. The residual function in C-.

The comments that appear after the signature are special comments for dcc to indicate which variables are dependent (here *yy*) and should be differentiated with respect to the independent ones (here *x* and *p*). Furthermore one can recognize that *yy* corresponds to the residual-function representing the dynamic model.

AC-SAMMM overloads the basic mathematical functions  $\sqrt{x}$ ,  $x^a$ ,  $(x/y)$ ,  $\log(x)$  and  $\exp(x)$  in order to increase robustness and/or efficiency of their numerical evaluation (see *acs\_pow(x,a,y)* in the above code fragment). An example is the mirroring of the square root at the origin. This is useful, for instance, when  $x > 0$  passes to  $x < 0$  immediately after a switching point, in case of a hybrid system. In order to detect a switching point precisely, a switching structure detection algorithm has to pass this point without switching being accomplished [3].

As a result of the script call the dynamic libraries called *car.dll* (in release mode) and *car\_d.dll* (in debug mode) are created. The intermediate C++ files are still available to the user but not needed to any further application of the model library.

The generated libraries are written in a neutral format so that they are not application-specific and can easily be exploited for a variety of purposes. We illustrate the application in the case of OCP, which can be resolved by means of direct single shooting, which has originally been introduced by Bruschi [2]. The basic idea is to substitute the control vector by an approximation (typically piecewise-constant or -linear), and to relax the path constraints on a grid  $t_0 < t_1 < \dots < t_n = t_f$ . This way the infinite-dimensional optimal control problem is approximated by a finite-dimensional nonlinear program (NLP). For this purpose we introduced a new C++ class called *MetaESO*. The aim of the *MetaESO* class is to define the constraints, the controls and the underlying time grids as input for the integrator. In addition, the user can define the sensitivities to be calculated. In summary, the user can define a restricted multistage OCP using the AC-SAMMM-generated dynamic library as dynamic model within this class (at present only: explicit switching times, simple box-constraints).

Using an integrator and an optimizer, the infrastructure is suited to solve a restricted class of multistage optimal control problems. The integrator used is NIXE (Hannemann *et al.* [6]). NIXE implements the extrapolated linearly-implicit Euler discretization for the solution of parametric differential-algebraic initial value problems (given by equations (3) - (5) of formulation **OCP**) and computes higher-order sensitivities by an efficient modified higher-order discrete adjoint approach or by forward sensitivity analysis. We show very briefly how to use AC-SAMMM as embedded dynamic model server, how to define the OCP and how to call the integrator NIXE. We omit details concerning the optimizer application and show the basic usage in Listing 5.

```

ACSAMMM_Eso* CarEso = new ACSAMMM_Eso( car );
MetaEso metaEso* = new MetaEso( );
metaEso->AddPhysicalStage( CarEso, 0.0, 1.0);

```

Listing 5. Dynamic loading of the model-library *car* in AC-SAMMM and usage of *MetaEso* class.

The second and third argument of *AddPhysicalStage()* represent the start and final time for the integration. As problem (10) has free final time, we had to reformulate:

$$x(t) = \Delta t f(x(t'), y(t'), u(p, t')), \quad t' \in [0, 1]$$

$$0 = g(x(t'), y(t'), u(p, t')),$$

where  $\Delta t = t_f - t_0$  is a parameter that can be controlled.

The constraints are defined using the *MetaESO* function *AddConstraint(stage Index, name, time, lB, uB, lagrMult)* as shown in Listing 6. The input *lagrMult* represents the value of the Lagrange multiplier corresponding to the constraint. This input could be delivered by an optimizer.

```

MetaEso->AddConstraint(0, velo, 1.0, 0, 0, lagrMult);
MetaEso->AddControl (0, accel, piecewiseLinear,
                    grid, parameters, -1.0, 1.0);

```

Listing 6. *MetaESO*: Constraints and Controls

In the same way the control parametrization of the acceleration could be the output of an optimizer. So the fifth argument of *AddControl(stageIndex, name, typeControl, grid, parameters, lB, uB)* is defined by the optimizer. The constraints define the states where sensitivities should be calculated. They are calculated with respect to the control-parametrization. The second-order sensitivities are calculated with respect to the Lagrangian function. Finally, as shown in Listing 7, the *MetaESO* object can be transferred to the integrator calculating the specified sensitivity information. More details about definition of related OCP see [13].

```
SecondOrderReverseHandler* sorHandler =
  new SecondOrderReverseHandler ( MetaEso );
sorHandler -> solve( );
```

**Listing 7.** *MetaESO* handler for second-order derivatives.

An interesting case study serving as a benchmark is a large-scale nonlinear system of about 2000 stiff DAEs [4]. It models the load change of an polymerization process. For real-time optimization (cf. Würth *et al.* [16]), about 40 second-order parameters and 160 first-order parameters of the model have to be computed. We measured the generation- time of the higher-order derivatives as well as the compile- and evaluation-times. All computations were performed on a Core2-Quad PC running Windows 7 on a 2.66 GHz CPU. The size of the dynamic library in release mode is about 6 MB.

Code generation time	Size of generated code	Compile time		Computation time of Hessian of Lagrange.
		with opt.	without opt.	
15 min	20 MB	hours	seconds	≈ 3 min.

**Table 1.** Computation effort for case study with 2000 stiff DAEs.

## 6 Summary and Outlook

We presented a platform for automatic algorithmic differentiation of mathematical models, which exhibits an object-oriented (extensible) interface that suits the needs to solve a differential-algebraic system. Included are `MOE2C`-, a Modelica to C code parser and `dcc`, the derivative code compiler for automatic differentiation.

A prototype-driver for multistage OCP exists serving as an interface between AC-SAMMM and NIXE. This interface will be developed further in future versions of AC-SAMMM. Also, further reductions of compile- and run-times can be expected. A major concern of the future development of AC-SAMMM is the proper treatment of hybrid systems [1], especially enabling discontinuity-locking and the handling of arbitrary complex switching conditions.

### References

- [1] Barton, P. I., and Lee, C. K. (2002). Modeling, simulation, sensitivity analysis, and optimization of hybrid systems. *ACM Trans. Model. Comput. Simul.*, 12, 256–289.
- [2] Brusch, R. G. (1974). A nonlinear programming approach to space shuttle trajectory optimization. *Journal of Optimization Theory and Applications*, 13(1), 94–118.

- [3] Cellier, F. E., and Kofman, E. (2006). *Continuous System Simulation*. Springer, New York.
- [4] Dünnebier, G., van Hessem, D., Kadam, J., Klatt, K.-U., and Schlegel, M. (2005). Optimization and control of polymerization processes. *Chemical Engineering Technology*, 28(5), 575–580.
- [5] Griewank, A., and Walther, A. (2008). *Evaluating derivatives: principles and techniques of algorithmic differentiation*. Soc. for Industrial and Applied Math. (SIAM).
- [6] Hannemann, R., Marquardt, W., Gendler, B., and Naumann, U. (2010). Discrete first- and second-order adjoints and automatic differentiation for the sensitivity analysis of dynamic models. In *Procedia Computer Science*, volume 1, pages 297–305.
- [7] Martin, K., and Hoffman, B. (2003). *Mastering CMake: A Cross-Platform Build System*. Kitware Inc..
- [8] Modelica Association, Linköping, Sweden. (2010). *Modelica - A Unified Object-Oriented Language for Physical Systems Modeling. Language Specification. Version 3.2*, March 2010.
- [9] Naumann, U. (2011). *The Art of Differentiating Computer Programs*. SIAM, 2011. To appear.
- [10] Naumann, U., Schenk, O., Simon, H., and Toledo, S., editors. (2009). *Combinatorial Scientific Computing*. number 09061 in Dagstuhl Seminar Proceedings, Wadern, Germany. Leibnitz-Zentrum für Informatik.
- [11] Pantelides, C., Keeping, B., Bernier, J., and Gautreau, C. (1999). *Open interface specification numerical solvers*. Technical Report CO-NUMR-EL-03 Version 1.08, CAPE-OPEN, 1999.
- [12] Schanen, M., Förster, M., Gendler, B., and Naumann, U. (2011). Compiler-based Differentiation of Numerical Simulation Codes. In *ICCGI 2011, The Sixth International Multi-Conference on Computing in the Global Information Technology*, pages 105–110. IARIA.
- [13] Schmitz, M., Gendler, B., and Hannemann, R. *Introduction to AC-SAMMM: A tutorial installing, and using AC-SAMMM*. RWTH Aachen, AVT.PT Process Systems Engineering, Aachener Verfahrenstechnik, Turmstraße 46, 52062 Aachen, to be made accessible.
- [14] Varnik, E. (2011). *Exploitation of Structural Sparsity in Algorithmic Differentiation*. PhD thesis, RWTH Aachen University, Aachen, Germany, submitted.
- [15] Varnik, E., and Naumann, U. (2009). What Color is the Non-Constant Part of Your Jacobian? In *[10]*, 2009, Extended Abstract.
- [16] Würth, L., Hannemann, R., and Marquardt, W. (2011). A two-layer architecture for economically optimal process control and operation. *Journal of Process Control*, 21(3), 311–321.

Submitted: July 2011 (ASIM STS Winterthur)

Accepted SNE: July 20, 2012

# Construction and Implementation of a Simple Agent-Based System on GPU-Architectures

Günter Schneckentreither<sup>1,2\*</sup>, Stefan Hepp<sup>2</sup>, Daniel Prokesch<sup>2</sup>

<sup>1</sup> dwh Simulation Services, Neustiftgasse 57-59, 1070, Vienna, Austria; \*[guenther.schneckentreither@dwh.at](mailto:guenther.schneckentreither@dwh.at)

<sup>2</sup> Mathematical Modelling and Simulation Group, Inst. f. Analysis and Scientific Computing, Vienna University of Technology, Vienna, Austria

**Abstract.** Agent-based modelling and simulation is still an upcoming approach for microsimulation. But a large number of agents with advanced dynamics and interactions requires sophisticated algorithms and lots of computational effort. We try to implement a rather simple but special agent-based model on GPU-architectures (graphics processing unit). This contribution presents the GPU implementations and investigates its practicability. Results show that implementation of agent-based systems on a GPU-architecture can deliver enormous speed-ups. Depending on the detailed structure of the system highly advanced algorithms deliver another great performance-boost. The combination of both approaches can deliver up to 1.000 times faster execution of the same simulation.

## Introduction

Agent-based models are usually applied when global behaviour of a system is best described by the micro- or macroscopic dynamics of a large set of similar but individual objects. These unique objects are called agents since their properties change individually and depend on the agents specific field of vision of the global system. A typical area of application is for example socio-economics. Every individual of a population can be represented by an agent with distinct features based on statistical data.

A large number of agents with advanced dynamics and interactions requires sophisticated algorithms and lots of computational effort. We try to implement a rather simple but special model on GPU-architectures (graphics processing unit), which are optimized for highly data-parallel operations. Our goal is to investigate the practicability and finally use such advanced programming techniques in order to increase the capacity of agent-based models.

## 1 Definition of the Test Model

The test model is a rather simple spatial system of moving agents. On a two-dimensional domain two sets of agents interact with each other depending on their distance. The first set are so-called ‘signposts’, whose positions are arranged at initialization and do not change over the course of simulation. Besides a fixed position the signposts hold a second static vector, which collectively defines a flow field on the domain. The second type of agents are referred to as moving agents since they hold two vectors which define their position and velocity.

In a simple scenario the evolution of the system is purely explicit and consist of two operations which are applied on the moving agents. The first operation changes the position of the agents according to their velocity:

$$\mathbf{x}(t + 1) = \mathbf{x}(t) + \mathbf{v}(t) \cdot dt$$

The second operation recalculates the velocity depending on the current velocity and the flow-field-vectors of ‘neighbouring’ signposts. The term ‘neighbourhood’ originates from cellular automata and defines a weighted set of interacting signposts for each moving agent. The neighbourhood-function can be for example a two-dimensional bell-shaped curve (normal distribution) or any other function, which describes a decaying likelihood of interaction.

Due to performance issues the neighbourhood-function may have a maximal radius for non-zero weights (compare search algorithms below). Once all weight factors  $\omega_{ij}$ ,  $j \in \{1, \dots, k\}$  for a moving agent  $a_i$  are calculated, the velocity changes according to an update function:

$$\mathbf{v}_i(t + 1) = \Phi(\mathbf{v}_i(t), \omega_{ij_1} \cdot \mathbf{v}_{j_1}(t), \dots, \omega_{ij_k} \cdot \mathbf{v}_{j_k}(t))$$

For agent-based systems the dynamics of different agents are usually mutually dependent. In this case the signposts would additionally change their properties according to a set of neighbouring moving agents. This type of coupled relation would for example occur if the motion of agents should obey or at least approximate physical flow of particles or a fluid etc., but is not focus of this investigations. To highlight the lack of such advanced properties we arrange a circular flow-field (flow-field vectors are tangential to the circular flow) and show that the moving agents drift from the centre to the boundary of the domain.

## 2 GPU-Architecture and Hardware

In the past decade a variety of new parallel computing architectures have been developed. Besides parallelisation on multiple CPUs (central processing unit), CPU cores or even physical machines it is possible to use graphics devices for performing calculations in parallel. This strategy is known as GPGPU (general-purpose computing on graphics processing units). Some of the most common architectures are CUDA (compute unified device architecture) for NVIDIA graphics devices [1], AMD FireStream for ATI devices [2] and OpenCL (open computing language), an abstraction layer for accessing the calculating capacity of multiple hardware devices [3]. The advantage of GPUs is the high throughput on data-parallel operations. On the other hand implementation is more complicated since the physical structure of the device is crucial for constructing the algorithm.

We use C++ and CUDA on a machine with the following specifications: 'AMD Athlon Phenom II 920' processor at 2.8Ghz, 3 GB of working memory and the consumer graphics device 'NVIDIA GeForce GTX 260' with 896 MB GDDR3 on-board memory using a 448-bit memory interface at 2.2Ghz, and a GT200b core with 24 streaming multiprocessors (SM), running at a core clock of 633Mhz with a shader clock of 1.4Ghz, providing Compute Capability 1.3 with CUDA Toolkit 2.3 and NVIDIA driver version 196.21.

A single SM executes one or more threadblocks consisting of maximally 512 threads [4, Sect. A.1.1]. Threads in a thread block are executed in groups of 32 parallel threads (thread warps) and execute the same instruction path. Executing different control flows within a warp (by predicated execution) reduces performance.

Major performance gain can be achieved through reducing and optimizing memory operations [5]. Several memory spaces are available within the CUDA programming architecture:

- **Host memory:** This is the RAM (random-access memory) used by the CPU only. Transfer of data to the GPU memory is rather expensive and happens through a PCIe bus with about 5 GB/s.
- **Global device memory:** This type of memory is attached to the GPU and provides a bandwidth of 50 to 80 GB/s. Copying from and to the (pinned) host memory can happen simultaneously to execution of code. Access to global memory is not cached and has a high latency, but this can be hidden efficiently by using a large number of threads.
- **Constants and texture memory:** This type of memory is cached by the SM and provides fast read access. We use textures for the signpost data and the acceleration structures since they are modified infrequently. The maximum size of a texture is limited to  $2^{16}$  by  $2^{15}$  texels for 2D textures, however this is enough to store up to  $2^{31}$  signposts.
- **Shared memory:** Every thread can use up to 16 KB of shared memory, which has a very low latency and allows communication between threads in a thread block but not different kernels.
- **Registers:** Every multiprocessor provides 8192 or 16384 32-bit registers in total which are shared between all threads of all thread blocks.

The number of available registers as well as the absolute maximum numbers for active warps, threads and blocks per SM depend on the 'Compute Capability' of the GPU.

The number of threads per thread block is further limited by the number of registers required for a thread block, which is determined by the number of registers used by the kernel multiplied by the number of threads per block and must not exceed the number of registers available on a SM.

The number of active blocks per SM is also limited by the sum of register and shared memory requirements of the blocks. [4, Sect. 4, 5.2, App. A].



### 3 Implementation on the GPU-Architecture

Additionally to the statements in Section 1 on coupled relations, exchange between the host system and the GPU are neglected. A simulation might require to store intermediate results on the hard-drive or to exchange data with another sub-model, which is not executed on the GPU. Such structures depend on the model and require balancing of computing time and accuracy [6]. The simulation is executed and visualized exclusively on the graphics device since we are interested in the basic performance of a GPU simulation.

There are two basic strategies for implementing the neighbourhood approach:

1. For each signpost, find all agents within the neighbourhood.
2. For each agent, find all signposts within the neighbourhood.

In the first case it is possible that multiple signposts try to update the velocity vector of the same moving agent, which would require performance decreasing synchronization mechanisms. Additionally, the second approach makes it more comfortable to implement optimized data structures for finding neighbouring signposts because their position does not change during simulation. Such data structures can be pre-calculated before the actual simulation and are discussed later on.

The following initialization steps are required:

1. Load or generate initial positions and velocities.
2. Generate the acceleration data structures for the neighbourhood search depending on the algorithms described below.
3. Copy all signpost and agent data to the GPU and initialize any constants and textures used by the kernels.

During simulating the kernel applies two operations per agent and time-step:

1. Find all signposts  $b_1, \dots, b_k$  which influence the agent  $a_i$  at position  $\mathbf{x}_i(t)$ .
2. Calculate the new velocity vector  $\mathbf{v}_i(t+1)$  and the new position  $\mathbf{x}_i(t+1)$  for agent  $a_i$ .

Performance of the second operation is determined by the number of memory reads required to collect the velocity vectors of all neighbouring signposts. This number ( $k \cdot C$ ) is not constant but bounded by the total number of signposts.

The time-critical operation is finding all signposts that influence a given moving agent. For this task several algorithms were implemented and compared.

#### 3.1 Linear Search

The simplest method for finding all signposts within a certain radius of an agent is to walk through the list of all signposts and compare the distance. Velocity vectors of appropriate signposts are added to the velocity vector of the current agent and then divided by  $k+1$ , where  $k$  is the number of neighbouring signposts (linear non-weighted influence).

The advantage of this approach is that all threads perform equal memory reads from the signpost array and thus can be cached very efficiently using texture memory. Furthermore, only a few instructions are needed to perform the lookup and there is almost no divergent control flow.

The only difference in control flow between threads arises when the influence of a signpost which is not within the neighbourhood region of a moving agent is ignored. On the other hand every agent needs to read and process each signpost in every step. The performance is therefore  $\Theta(n \cdot m)$  for  $n$  agents and  $m$  signposts. Even for small  $m$  this method is slower than using a uniform grid.

#### 3.2 Uniform Grid Search

In order to speed things up the domain is divided into a regular grid. The cell index of an agent or a signpost can quickly be determined without any memory lookups (except for the constants specifying the grids structure). Prior to the simulation the list of all signposts is sorted by cell index so that signposts of the same cell are stored consecutively. An additional index array stores the index of the first and the last signpost of each grid cell.

If the cell size is greater or equal to the neighbourhood radius only signposts in the same cell as the agent and in at most three adjacent cells can affect the motion of the agent. The adjacent cells which need to be checked can be determined by comparing the distance of the agent to the borders of its cell with the neighbourhood radius (no lookups into global memory except for cached constants).

To calculate the new velocity of an agent (from its cell index) the index of the first and the last signpost in each of the four 'neighbouring' cells is taken from the index array.

The signposts which actually influence the agent are then searched in the four resulting lists of signposts using a linear search as described above.

If there are at most  $k$  signposts in each cell, at most  $4 \cdot k$  signposts are checked for every agent in every step, therefore the performance of this algorithm is of order  $O(4n \cdot k)$  for  $n$  agents.

This method works best if all signposts feature the same neighbourhood structure (as it is the case in this simulation) and if they are distributed uniformly without areas of strong aggregation. Generally a dense distribution leads to a larger value for  $k$ , whereas a sparse distribution can lead to empty cells and memory consumption in the index array.

If the grid diameter is smaller than the neighbourhood radius, more cells have to be checked for neighbouring signposts but fewer signposts must be neglected. However, since every agent still needs to find and process all signposts within its radius, a very small cell size cannot reduce the number of memory reads below the number of signposts affecting the agent but introduces a larger overhead for index array reads.

Assuming that the influence of the signposts does not depend on the distance between the agent and the signpost, a smaller grid size could still improve the performance for dense distributions. For every cell the velocity vectors of all signposts which completely overlap the cell can be accumulated and stored in advance, thus reducing the number of individual signposts needed to be checked during simulation.

The algorithm to sort the signposts and create the index array can also be implemented efficiently with CUDA using the following algorithm (compare [7]):

1. Allocate a memory array for the index array.
2. For every signpost, calculate its grid cell index based on its position. Clearly this can be done by starting a kernel for every signpost.
3. Sort the signposts regarding to their cell index. This can be done on the GPU e.g. by using the radix-sort implementation of the CUDA Thrust library.
4. For every grid cell, find the index of the first and the last signpost in the sorted signpost array. This can be done by starting a thread for every signpost.

Since the index calculation can be done entirely on the GPU, this algorithm can also be used to handle moving signposts or to let the agents interact with each other.

### 3.3 Balanced Tree Search

To tackle the memory overhead for non-uniform signpost distributions, a third search method was implemented using cells with variable size.

Instead of a uniform grid a kD-tree is constructed so that each leaf has the same depth and each subdivision of an area is done so that the two new areas contain roughly the same number of signposts while keeping the cell size larger than the signpost diameter.

The algorithm is similar to the uniform grid search method, except that finding an agents cell requires  $\log_2(m)$  memory read operations to descent the kD-tree, where  $m$  is the overall number of cells. Again adjacent cells need to be checked if the distance of an agent to the cell border is smaller than the neighbourhood radius. This search is more complex but can be pre-calculated for every cell.

Early tests have shown that even if the signposts in adjacent cells are not checked, the cell search algorithm leads to a decreased performance compared to the uniform grid approach. The only benefit could be that in a non-uniform or sparse distribution fewer cells are needed to get the same maximum number of signposts per cell. But even a 2048 by 2048 cell uniform grid, which is more than enough for our purposes, requires only about 33Mb of additional memory for the index array. Therefore the implementation of this search method was not completed in favour of the uniform grid search.

### 3.4 Caching and Other Search Algorithms

A cache could be used to reduce the number of signpost-checks. The only memory space where a persistent cache between two kernel executions can be implemented is global memory. The cache must have a fixed size because resizing allocated memory is not possible during kernel execution and allocating memory is a costly operation.

Consequently implementing a cache is beneficial only if accumulated influences can be calculated for multiple time steps and the resulting information does not require too much memory space. Since signpost influences can vary frequently it is difficult to calculate and store useful cache data without much overhead. In sparse distributions only few signposts need to be checked. Thus even a cache hit can result in more memory operations compared to an implementation without cache.

Other alternatives to search for signposts are a k-Nearest-Neighbour (k-NN) search or a range search combined with storing the signposts as nodes of a kD-tree. However this has similar advantages and drawbacks as the balanced tree search; additional index arrays are not required but an additional  $\log_2(m)$  overhead in memory reads and a stack for the tree lookups is required.

This could be implemented using bit-arrays which can be stored in registers or shared memory to avoid costly global memory access if the traversed tree is a binary tree and the maximum tree depth is fixed at compile-time.

The control flow and the memory access pattern can diverge greatly between consecutive threads, which has a negative performance impact for CUDA kernels. Therefore the benefit of such more complex algorithms compared to the uniform grid search is a lower memory requirement for large and sparse datasets and the ability to search for arbitrary ranges at the cost of more complex algorithms which require more control flow, arithmetic and memory operations. Arbitrarily deep trees cannot be handled due to the lack of dynamic data structures in the CUDA architecture.

As already mentioned before, both signposts and agents can be sorted by a grid cell index on the GPU. If this is done for every simulation step, this can be used for several things. If the signposts are sorted, they can be moved around while keeping the acceleration data structures up-to-date. If the agent array is sorted, agents can interact with each other efficiently.

An additional benefit of sorting the agents is that consecutive agents will usually fall into the same grid cell and therefore threads calculating the next step of consecutive agents will have a similar control flow and perform similar memory reads which the hardware is able to coalesce into a single read or at least increase the likelihood for a cache hit for texture memory reads.

As an additional optimization data required by all threads for agents in the same cell like the list of signposts in this cell can be cached in shared memory, thus speeding up random reads of the same data by multiple threads. However the performance impact of this optimization has not been tested.

## 4 Further Technical Details and Visualization

For debugging and comparison purposes, two simulation kernels have been implemented. The first implementation uses the CPU exclusively, the second kernel transfers all data to the GPU memory at initialization and performs the simulation including visualization entirely on the GPU. If the CUDA simulation results should be written to disk, the agent array needs to be written back to the host memory for every frame which should be stored.

During development and for analysing results of a simulation – especially in the case of a spatial model – it can be very useful to have a decent front-end for visualizing the dynamics of the system. Such a front-end was build using OpenGL. Several features of the system can be displayed or masked during simulation. The signposts and optionally the grid are pre-rendered once into a texture using an OpenGL frame-buffer. For the agents, three different modi have been implemented.

Moving agents are either rendered as soft points with three pixel radius using OpenGL and supplying the coordinates and colour of the agents as a vertex array, as simple triangles, again using OpenGL and vertex-arrays, or by drawing the agents as single pixels into a texture buffer and displaying the texture. In all three modi, the data is written either by the CPU to a memory-mapped OpenGL buffer, or – if CUDA is used – directly from graphics card memory to an OpenGL buffer in the graphics card memory using a CUDA kernel and the CUDA OpenGL interoperability methods.

In the render-to-texture mode a simple scatter algorithm is used, i.e. for every agent a kernel is started which writes the pixel at the agents position. This can result in multiple writes to the same pixel, where the order of the writes is random for the CUDA implementation, which can lead to rapid colour changes between frames (on the CPU, all agents are processed in serial, so there are no conflicts due to concurrency).

A better approach for the CUDA implementation would be a gather algorithm, similar to the algorithm used to create the index array of the uniform grid search. However, since the performance of the OpenGL implementation using vertex-buffers is as fast as the current render-to-texture implementation, the more complicated gather algorithm was not implemented.

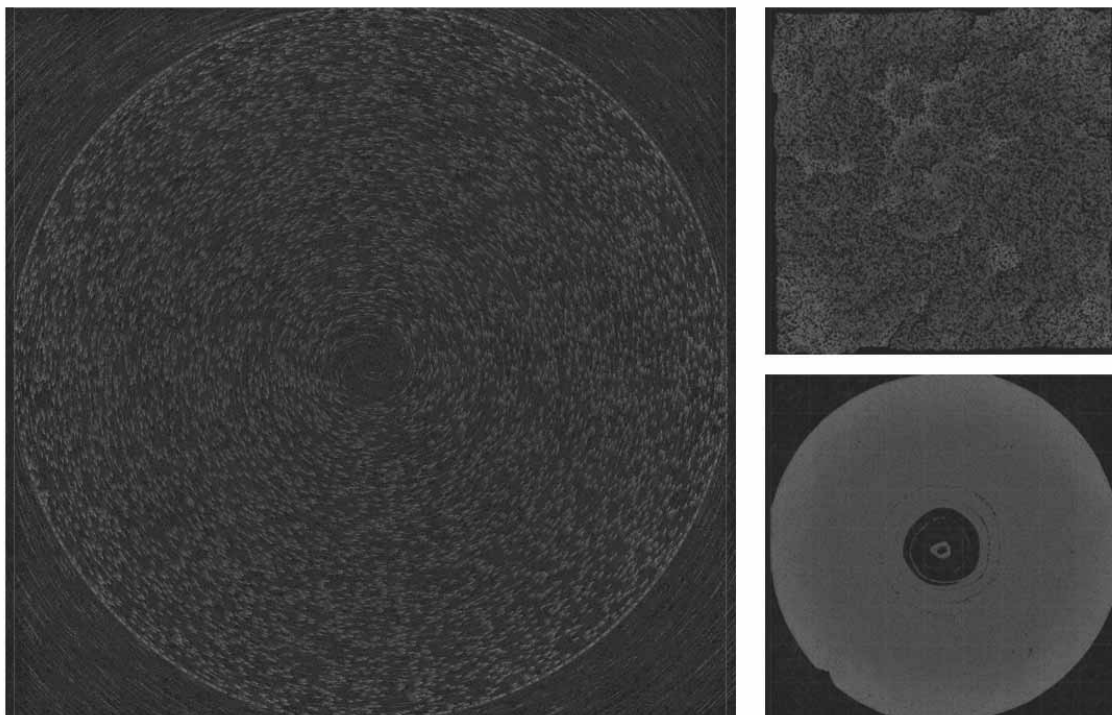


Figure 1. Few agents represented as triangles (left) and 1,000,000 agents drawn as pixels (right).

The speed of the agents is visualized using the colour of the points or triangles, ranging from red for slow agents to green for fast agents (Figure 1).

Agents	CPU			GPU		
	Points	Triangles	Texture	Points	Triangles	Texture
10,000	1.5	2.0	3.5	2.1	2.2	2.3
1,000,000	90.0	220.0	70.0	8.0	18.0	7.0

Table 1: Additional overhead needed for rendering.

## 5 Results

To measure the performance of the implementation, the average computation time per step was measured. For all measurements, the same setup was used: The signposts are placed randomly in a rectangular field directing the agents into circular motion, the agents are placed randomly on the domain and boundary conditions are reflective. Visualization has been disabled for these benchmarks.

Using a uniform grid data structure with 100 cells increases the performance even for a relatively low number of signposts (Figure 2). Between the CPU and the GPU implementation, speed-ups between 70 times to 100 times can be observed for the linear search approach. If a uniform grid is used, the GPU is about 50 to 70 times faster than the CPU implementation. As expected the simulation time is linear in the number of agents. For CUDA, a small constant overhead can be observed for less than 100,000 agents.

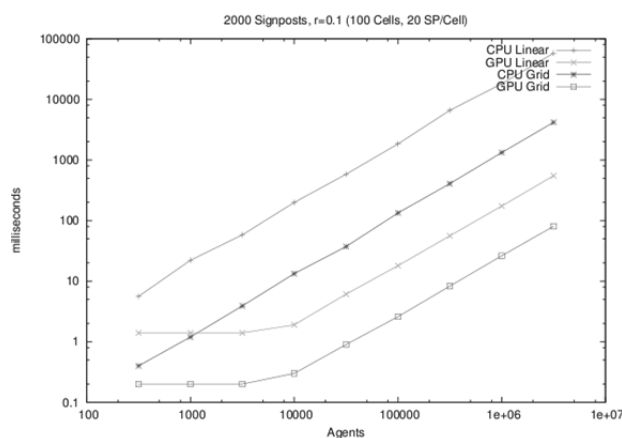


Figure 2. Calculation time depending on the number of agents.

Agents	Signposts	Radius	CPU (ms)	GPU (ms)	Signposts/Cell	Speed-up
$10^5$	$10^5$	0.001	33.8	0.5	0.025	68
		0.0032	37.0	0.7	0.250	53
		0.01	93.6	3.8	2.500	25
		0.032	603.7	30.1	25.000	20
		0.1	5224.0	213.1	250.000	25
$10^6$	$10^6$	0.001	652.5	10.3	0.250	63
$10^7$	$10^6$	0.001	4966.8	187.0	0.250	26

Table 2. Performance depending on number of agents and neighbourhood radius.

Observing the average time per simulation step over the number of signposts exhibits the same linear behaviour.

For the linear search again a speed-up of about 100 times can be achieved by CUDA. For the grid search the speed-up is again lower between a factor 10 and 20.

For the linear search approach all kernels perform the same memory reads in the same order and therefore can be coalesced and cached more effectively than for the grid search approach which issues memory reads in a more random fashion (this could be tackled by sorting the agents by their grid cell index).

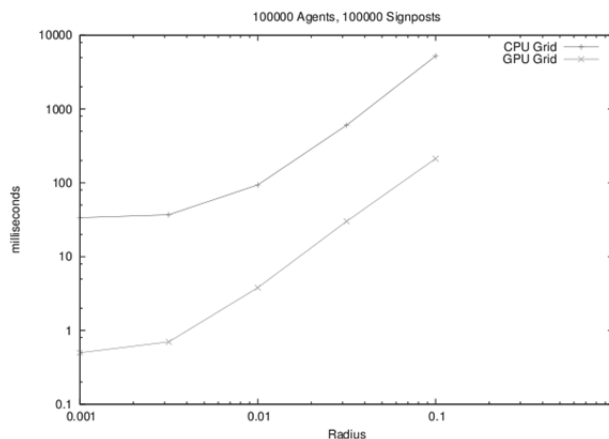


Figure 3. Calculation time depending on the neighbourhood radius for a large number of agents.

When a grid is used with  $k$  cells to speed up the signpost search for  $m$  signposts, only  $\frac{4m}{k}$  signposts will be checked on the average per step by every agent instead of all  $m$  signposts.

This leads to a speed-up of the simulation (except for a higher memory access penalty due to more random memory reads).

If the grid cells are made as small as possible, i.e. the diameter of the neighbourhood, the simulation time increases quadratically with the signpost diameter and linearly with the number of signposts, as shown in Table 2 and Figure 3.

Development (familiarizing with CUDA and implementation) took about 6 man-weeks for advanced programmers. The additional CPU-only implementation imposed nearly no additional overhead since the code for the CPU implementation and the CUDA kernels is quite similar, and simplified debugging the algorithms.

## 6 Conclusions

Results show that implementation of agent-based systems on a GPU-architecture can deliver enormous speed-ups. Depending on the detailed structure of the system highly advanced algorithms as discussed in Section 4 deliver another great performance-boost. The combination of both approaches can deliver up to 1,000 times faster execution of the same simulation.

Of course the applicability of both GPGPU and advanced search algorithms depends on the model itself. In general every model requires distinct implementation techniques, which are often more complex than a straight-forward implementation. However, depending on the structure of a model higher implementation effort can be very profitable.

The test model inhibits a unidirectional structure, which is predestined for applying highly parallel techniques. But also large spatial agent-based models with coupled relations can be improved using a (uniform) grid approach or search trees.

Depending on the interaction of agents static acceleration data structures can be used, but also dynamic look-up tables can be more efficient than a linear search approach.

The crucial performance gain for all approaches can be achieved by optimizing memory access.

## References

- [1] [www.nvidia.com/object/cuda\\_home.html](http://www.nvidia.com/object/cuda_home.html)
- [2] [ati.amd.com/technology/streamcomputing](http://ati.amd.com/technology/streamcomputing)
- [3] [www.khronos.org/opencvl](http://www.khronos.org/opencvl)
- [4] 'NVIDIA CUDA Programming Guide 2.3.' NVIDIA Corporation, October 2009.
- [5] 'NVIDIA CUDA Best Practices Guide 2.3.' NVIDIA Corporation, July 2009.
- [6] Stam, J. (2009). Maximizing GPU Efficiency in Extreme Throughput Applications. Video Presentation presented at the NVIDIA GTC09, October 2009.
- [7] Tonge, R. (2009). . Spatial Data Structures for Massively Parallel Computing. Video Presentation presented at the NVIDIA GTC09, October 2009.

Submitted: March, 2011

Revised: July 15, 2012

Accepted: October 10, 2012

# Physical Modelling in MvStudium

Yuri Kolesov, Yuri Senichenkov\*

St. Petersburg State Polytechnical University, Polytechnicheskaj 29,SPU,195251 St. Petersburg; \*senyb@dcn.infos.ru

**Abstract.** In this paper, we describe ‘physical’ modeling technology in MvStudium that is a graphical environment for modeling and simulation of complex dynamic systems. MvStudium.6 modeling language (MVL) is based on open hybrid automata or Behavior-Charts (for short B-Charts), functional diagrams, and supports object-oriented modeling. B-Chart is an extension of UML state machine with do-activities in the form of differential-algebraic equations. External variables of MVL classes may be in the form of ‘Inputs-Outputs’ (I/O) or ‘Contacts-Flows’ (C/F). C/F components in MvStudium are analogous to Modelica components but in MVL there is no limitation on type (NAE, ODE, DAE), dimension, and form (explicit, semi-explicit, implicit) for a current solved system of equations prescribed to a current state of a B-Chart. MvStudium compiler automatically builds an executable model. It may run under the environment or it may be built in the form of DLL and used even in real-time applications.

B-Chart is an extension of UML state machine with do-activities in the form of differential-algebraic equations without History and Orthogonal states. It is used in MVL for describing hybrid behavior and planning computer experiments.

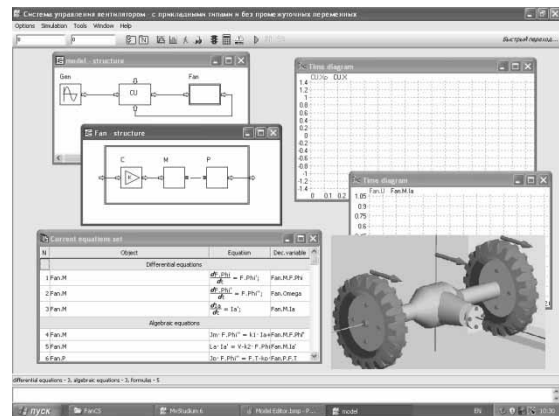


Figure 2. MvStudium virtual test bench (ModelViewer).

## 1 General

Graphical environment MvStudium.6 [1,2,3] for modeling (Model Editor) and simulation (Virtual Test Bench) of complex dynamical systems is based on formalism of open hybrid automata or Behavior-Charts (for short B-Charts) (Figure 1,2).

MvStudium.6 supports object-oriented modeling: MVL classes and packages make it possible to reproduce hierarchical structure and complex behavior of real world system (physical modeling). External variables of MVL classes (model components) may be in the form of ‘Inputs-Outputs’ (Figure 3) or ‘Contacts-Flows’ (Figure 4).

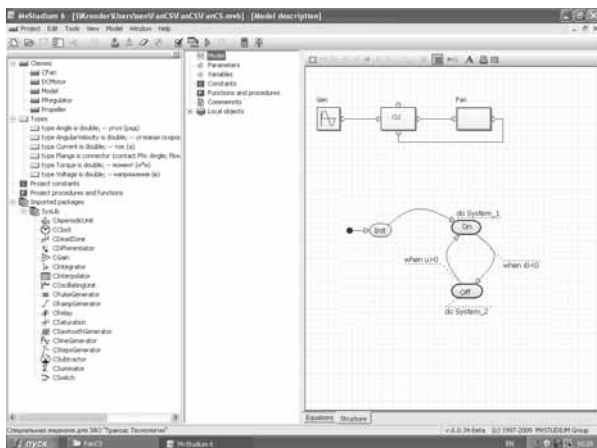


Figure 1. MvStudium model editor: a functional diagram and B-Chart.

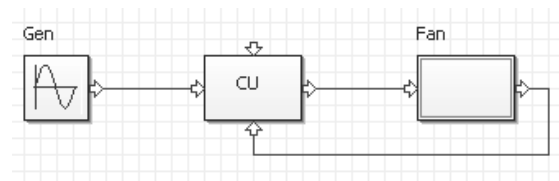


Figure 3. I/O components.

Physical modeling or using objects with “Contacts-Flows” and B-Charts requires rebuilding of a solved system of equations after any change event resulting in changing of a local behavior of a B-Chart. Rebuilding assumes structural analysis and reducing of a new system on run-time.

Building a new final system often leads to underdetermined differential equations especially on early stage of designing a new model.

It is not quite clear, should we consider such systems as user's error or we need to help him and automatically build a determined differential equations (if possible).

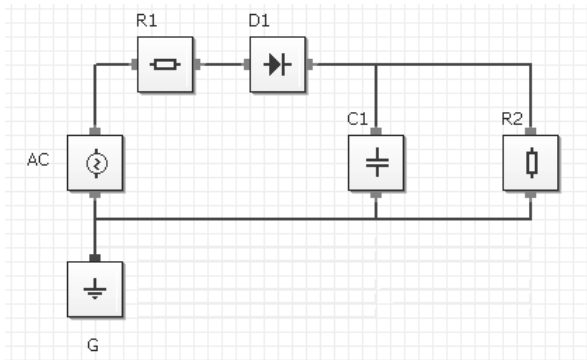


Figure 4. C/F blocks.

It is very important for 'physical' modeling to decrease as much as possible dimension of final system of equations. For example we can delete from it contact and flow equations, solve some equations symbolically, or divide algebraic equation on non-linear and liner parts and consider linear equations as substitutions in Newton method resolved by Gauss elimination. For this purpose a special module for symbolic calculations was developed.

## 2 Object-oriented Modeling in MvStudium

MvStudium's classes may be divided into four groups (stereotypes) that are: a) isolated classical dynamical systems, b) isolated hybrid systems, c) multi-component models with 'Input/Output' and/or 'Contact/Flow' components, d) hierarchical multi-component models equipped out its own B-Chart for planning virtual experiment. A class [2] may have attributes: external variables (links), structure and behavior, and it may inherit attributes of its parent. A special class MODEL has only one instance and it is compiled in executable program.

MVL uses B-Chart for describing complex event-driven behavior. Graphical notation of B-Chart uses elements of UML state machines and activity diagrams. It is forbidden to use History and Orthogonal states. However the main difference between UML state machine and B-chart is in interpretation of do-activities. In MvStudium do-activity is a solution of a system of algebraic-differential equations.

Do-activity is a local class with stereotypes 'continuous' or 'hybrid'.

## 3 Model Stereotypes

There are four model stereotypes. A MvStudium's model stereotype that forms a tree (Figure 5) with the parent is called 'continuous'.

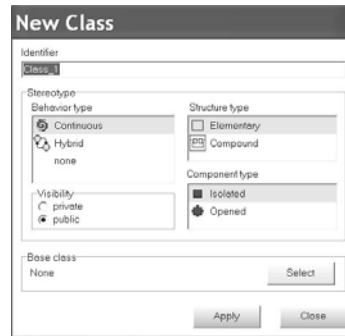


Figure 5. 'New class' dialog.

'Continuous' [3] stands for an isolated system written in the form of a system of algebraic-differential equations:

$$\begin{aligned}
 F(t, x, \frac{dx}{dt}, \frac{d^2x}{dt^2}) &= 0 \\
 x(t), F &\in \mathfrak{R}^n, \\
 x(0) &= x_0
 \end{aligned}
 \tag{1}$$

Equations may be written in scalar or vector forms. A special Equation Editor allows writing and editing equations in a usual mathematical form.

The second stereotype is called 'hybrid'. This stereotype uses UML state machine notation for describing hybrid systems or systems of algebraic-differential equations with discontinuous right sides. A 'continuous' class may be automatically transformed into a 'hybrid' class. In this case 'hybrid' class will have a B-Chart with one node and one local class. For editing a hybrid class, a user needs additionally a B-Chart Editor.

The third stereotype is used to design multi-component models using a common for an engineer 'functional diagrams' graphical image. Classes used in a functional diagram may have external variables of the following kinds: 'Input/Output' (I/O) and/or 'Contact/Flow' (C/F).

Using blocks with I/O and B-Charts does not create any difficulties for building a final system of equations for a current state of a MODEL.



A code for a final system for a current Model state may be built automatically from codes of local class systems compiled beforehand. "Correct" local equations lead to 'correct' final system.

In the case of classes with C/F and B-Charts we deal with underdetermined local systems of equations and we can speak only about 'correctness' of a final system for a current Model state. There is no need of checking 'correctness' of all possible final systems beforehand. It is quite reasonable to analyze only 'correctness' of realized on run time final systems (particular trajectory of hybrid automaton). For this stereotype it is necessary to use additionally a Functional Diagram Editor.

The forth stereotype called 'Functional diagram with control B-Chart' is used for complex computer experiments. A functional diagram is equipped by control B-Chart. While using a B-Chart for planning experiment it is possible to prescribe to a state behavior of a model component.

Model stereotypes make it possible to support 'natural' technology of designing 'from simple to complex'. As the first step you realize isolated model in the form of differential equations disregarding structure and different modes (stereotype 'Continuous'). As the second step you design event-driven model (stereotype 'Hybrid').

Then you take into account structure (stereotype 'Functional diagram') and start planning experiments with the whole model ('Functional diagram with control B-Chart'). This simple technology works well for new scientific problems, but for industrial design it is preferable to use the UML design technology.

### 4 Physical modeling in MvStadium

Using C/F components in MvStadium does not differ from using similar components in Modelica (Figure 6, 7, 8, 9).

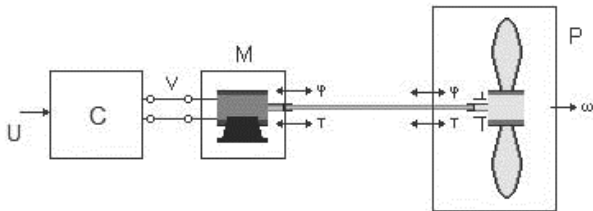


Figure 6. Control system for 'Motor and Propeller' is well known Modelica's example of 'physical' modeling.

C/F components in MvStadium are analogous to Modelica components but in MVL there is no limitation on type (NAE, ODE, DAE), dimension, and form (explicit, semi-explicit, implicit) for a current solved system of equations prescribed to a current state of a B-Chart. Using information about an 'old' system MvStadium builds, analyzes structure and finds consistent initial conditions for a 'new' system. An 'old' and a 'new' system may have different unknowns, dimensions, types and forms. We do not know conditions guarantying against appearance of underdetermined final systems for all admissible behaviors of a hybrid automaton composition with given local behaviors.

Our experience shows that underdetermined final systems more often appear during conceptual stage of new model designing. Automatic building of a determined system in this case is a cost-based and dangerous operation resulting sometimes in not physically feasible behavior. What is more it is very difficult to distinguish a very similar behavior from a true one.

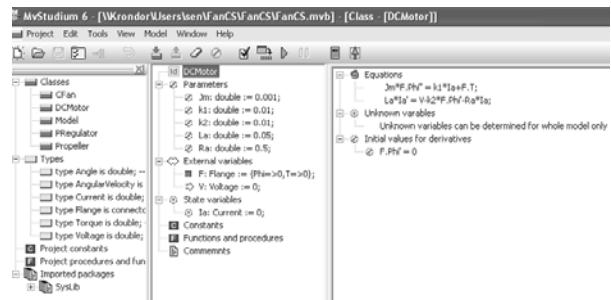


Figure 7. Controller.

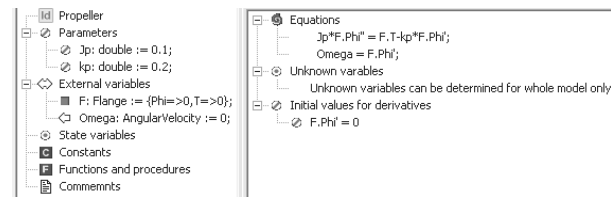


Figure 8. Propeller.

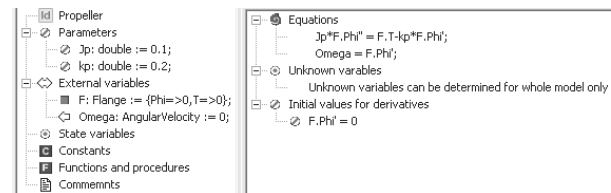


Figure 9. Motor.

In the present version of MvStudium a final system is built using topological equations (structure of functional diagram) and component equations (local behavior of components). MvStudium detects underdetermined and high index final systems (using Pantelides' algorithm) and warnings for a user. For finding consistent initial conditions different modifications of Newton's methods are used. Numerical methods are divided into three groups: numerical methods for solving non-linear algebraic equations, ordinary differential equations and algebraic-differential equations. Numerical solution MvStudium for ODE and DAE starts initially with an explicit method and if stiffness is detected an explicit method is changed for an implicit one automatically (Figure 10).

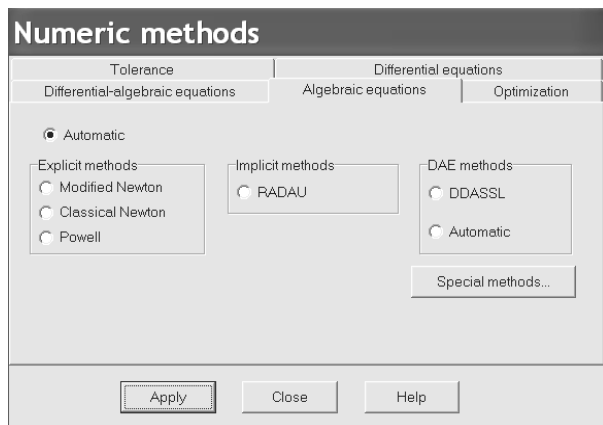


Figure 10. Numerical methods.

### 5 Visual Debugging

Modern graphical environments for modeling and simulation copes well with building and solving final systems. However the main problem is not in automatically building of a model, but in proving that the built model corresponds to a real world system. In the present version of MvStudium a user can trace a process of forming and solving equations and watch a built final system in a usual mathematical form. It is not enough for a large scale system, but better than nothing.

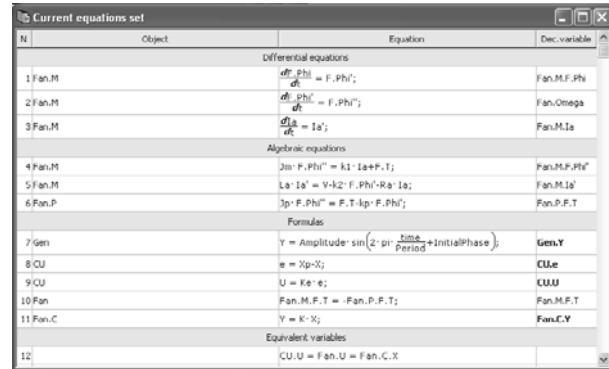


Figure 11. Virtual Test bench: a current final system.

### 6 Symbolic Calculations

One possible way of increasing effectiveness of simulation is to use symbolic calculations to transform and simplify the final system. Symbolic calculations in MvStudium are used for differentiation of expressions and equations, solving some types of equations (when it is not expensive), simplification of equations, decreasing the dimension of the solved numerically final system.

#### References

- [1] Breiteneker F., and Popper N. (2009). Classification and evaluation of features in advanced simulators. *Proceedings MATHMOD 09 Vienna*, Full papers CD Vol.
- [2] Kolesov Y. B. (2004). Object-Oriented Modeling of Complex Dynamical Systems. *St. Petersburg: Publishing House of the Polytechnic University 239*.
- [3] Senichenkov Y. B. (2004). Numerical Modeling of Hybrid Systems. *St. Petersburg: Publishing House of the Polytechnic University 206*.
- [4] Altunin K.Y., Kolesov Y. B., and Senichenkov Y. B. (2009). A tool for modeling and simulation of complex dynamical systems. *Proceedings of the distributed intelligence systems and technologies. Workshop DIST, St. Petersburg, 2009, Russia*.

Submitted: March, 2012

Revised: July 15, 2012

Accepted: November 10, 2012

# BroSAnT – an Example of the Mechatronical Way of Thinking

Sergey Petkun\*, Xin Zhao

Brose Fahrzeugteile GmbH & Co. Kommanditgesellschaft, Hallstadt; \*[Sergey.Petkun@brose.com](mailto:Sergey.Petkun@brose.com)

**Abstract.** The desire of the modelling of mechatronics system is daily growing. But already at the formulation of aims such simulation there are many different meaning what should deliver system simulation as resulting output. It is connected together with system responsibility on whole system. Mechatronics systems are naturally domain distributed systems and the main interest on system simulation is at the acquisition phase (concept assurance) and at the final phase (functional application the whole system). At the beginning the question is 'Which components should be taken?', at last phase will be asked 'Why it doesn't work how we have imagined it?'. The question, usually coming from management 'How can I spare money with system simulation?', should be transferred in another one 'Can I reach the required quality without system simulation?'. Very often by system modelling there is not enough information about components at project beginning, at the end of it - vice versa: too much information. The next system simulation feature is in it's naturally multi domain character. The boundary condition between domains is known problem almost in every companies. As consequence for system simulation is its membership and place in simulation infrastructure remains open. The system analysis has character of multi criteria because the model behaviour will be estimated from different point of views: mechanical, electrical and etc. from such point of view, BroSAnT (Brose System Analyse Tool) tries to realise the new conceptual solution. The origin of false thinking is the meaning that the system modelling can and should be made like simple joining the modelling of specific domains. In our opinion, only from system view, the necessary modelling level of corresponding domain counterpart can be right determined.

## Introduction

Everybody, who tries to simulate the whole system, has own imagination what the simulation should do. In reality, the system simulation is concentrated with Know-How about the whole system. Under whole system, is meant some mechatronics system: system, consisting of electrical power source, energy transformer (electrical into mechanical), mechanical executive me-

chanism and intellectual control system (algorithms realized in software and controlling hardware). There are a number of approaches to simulate the components: solid multi bodies in mechanics like Adams or Simpack, Matlab-Simulink in Algorithms/Software, Spice/VHDL-AMS modelling in hardware.

If there are some models of different components in system, the first idea that comes is to combine these available models in whole system. It leads to co-simulation concept. There are a lot of techniques for co-simulation but it is not an optimal way for system analysis. The system analysis should begin from question 'is the system appropriate to its purposes?' During developing mechatronic systems, it is highly recommended to optimize appropriation to constitutive components. Common functionality of whole system is connected with functionality of every component: if the system does not right work – from system point of view, it is not so important which component is defect – the result is clear: the device is defect. It doesn't matter whether electronics fails or mechanics breaks down.

There is a false conviction that with help of advanced software the constructional errors can be corrected. The poor constructed mechanical structure can be brought to functional state, but compelled solution is not the best solution. The optimization on system level has a number of different and very often contradictive criteria. This contradictory consists in reducing price of component on one hand and on the other hand in providing the required functionality. It is not seldom case when the device is defect only because of some trifle fails.

## 1 Functionality under 'All Conditions'

Requirements, described in specification, predestine the choice of parameters for mechatronics simulation: all working condition: climatic, constructive, dynamic and energetic (e.g. voltage fluctuation on the power source).

It is clear that the whole range of all parameter variations can not be investigated for different reasons but it is known (and partly it is company know-how) which effects should be taken into account during modelling components.

All these questions to simulation can be yet further extended or reduced, depending on situation. For example, for some industrial project with huge number of mass production, the constructive tolerances take great importance which is not so interesting in the case of some single sample. The most of mechatronic systems have some 'adjustable' part in form of algorithms parameter set (firmware parameter part), which allows optimizing the system for certain 'working range'.

To summarize all these requirements on mechatronics simulation, all input variation can be divided in groups:

- Constructive setting
- Climatic environment
- Dynamic environment
- Energy supply
- Algorithmic setting

The purpose of system is usually described in specification on system. The working environment is usually prescribed in the specification. For mechatronics system, it is, as a rule, ambience, dynamic and constructive tolerances. Under all these conditions, the system should be complied with the purpose determined in specification.

From this list, it can be seen that there are already enough parameters only at system level without taking component level into account. That is why, some flexible cable, working far from large deformation range, can be treated as only force transfer element with certain elasticity properties. So the complex constructive element will be reduced to some functional property as force transfer with definite stiffness.

Such approach is well known since a long time through modelling the electronic schemas where complex electrical behaviour of transistor will be replaced by behaviour model at working point. From system point of view, mechanics is nothing else as force-velocity transformer in dynamical system.

From view behavioural modelling, the whole mechanical effects can be described as combination of elastic, inertial and frictional properties, describing these effects. Elasticity is responsible for recoverable energy transfer, inertia - for time delay by the movement, frictional for energy loss. Naturally, all these properties strongly depend on conditions, mentioned above, but requirement in specification is also based on certain 'working range'.

Replacing the complex mechanical geometry through behavioural description is not simple task but brings enormous advantages with it.

To provide industrial projects with CAE support, the simulation should be 'real time' able in sense of, not only fast calculation but also system building, overview and analysable. Nobody needs simulation results after successful start of production. At present, the development of mechatronics system is connected with combination of prefabricated hardware components (electrical, mechanical) available on commercial market and that is why the importance of simulation at system level rises continually. The lack of domain compatible models leads to the development of new modelling languages as Modelica [1] or effort to describe the wanted components of system in term of old one as VHDL-AMS [2]. There is already first effort in standardisation of models on component level [3]. The electronics development sets a good example for future technology of development - the product is impossible to sell without accompanying model with it. The risk of implementation of inappropriate component will be reduced through virtual assembling test.

Successful and reasonable simulation of whole system is not possible without participation of domain experts in it, because the simulation engineers can not right estimate if the behaviour of the whole system reflects all necessary effects under investigation. The aim of simulation developers is to give appropriate component models with minimal number of parameters but with all necessary effects in it. It leads to the well-known solution: server-client application.

There are not so many problems connected with embedded control algorithms, because usually using platform specific C-code at end application phase allows to bind this code directly as 'black box' in simulation model.

Almost all simulators can use the 'black box' with little adaptation by embedding it in simulation model. This feature is very important for system simulation because it give us some so necessary criteria for estimation how close the rest of model is to reality.

Very important point of system simulation is the ability of model verification. The focus of system analysis is dynamical behaviour, and that is why, the comparison with real measured behaviour must be possible. What can not be measured is not well appropriate for the estimating the quality of model.

The traceability of simulation runs is obligatory by system simulation. The number of parameters, even in the case of simple simulation, is not small and the number of possible investigation variants is very big.

There are some different meanings about GUI (Graphical User Interface) for end user. In our opinion, the same simulation interface should be adapted to end user for specific domains. By analyzing the same system, the experts from different domains have different points of interests to the same object and various analyzing methods: for mechanics the forces are in the focus of analyses, current or voltage can be in the focus of analysing for electronics.

System simulation is in certain sense also documentation for making some conceptual decision. That means the simulation results should be reproducible in a later time point. The requirements to system simulation depend also on simulation users. System developers have other demands to simulation as system application team.

## 2 System Components

To divide system into components is not complex - it is determined by construction of product. The constituent part of product can be used from self manufacture or bought on the commercial market. The most important feature is that this component can be non-destructively replaced in device. The second one is that the part should possess some self-completed functionality.

The system analysing tool should be based on some data base with component models. It determines the choice of modelling language. It can be some standard language as VHDL-AMS or another popular one as Modelica or at all tool specific language like MAST for simulator Saber (Synopsys).

The modelling of components is very important to have, if it is possible, the similarity to reality by choosing connection points. For example, for drive there are three connection points: two electrical and one mechanical. It gives possibility to use different drive models and exchange them in a light way. It does not matter if the used drive with another parameter set or at all some other models, maybe with another modelling depth. From system point of view, a drive has the main functionality to transform electrical energy into mechanical movement.

This sameness allows the product developer to use different drives by simple replacement of component. Already at modelling level the interface between system components should be clear determined.

There is big risk during determining the depth of component description to go down too deep in the component description. Component level should not be mixed up with system level. Domain experts are trying continually to deep in each own domain. The practical rule is following: main parameters, which are needed for component description on system level, can be found in the specification sheets of this component. It stands to reason that, if the number of parameters is not enough for self-consistent description of component, it means that this specification gives component supplier more freedom and can lead later to problems on the side system responsibility. It is well-known case: the component is complied with its specification but system with it does not work right.

It is obvious that every additional requirement in specification is connected with price or opposition from supplier side but the depth of specification on the product shows 'gold' practical middle between customer and supplier and also reflects current industrial state in this manufacturing branch.

Electronics development is leading in development based on CAE techniques and it can be useful to transfer huge practical experience that this domain already has, in other domains. As evidence, it may be mentioned the rate of increasing amount of transistors in modern CPU.

## 3 BroSANT Objectivities

The development of **BroSANT** is based on multi purpose basic:

- Personal education
- Keeping company know-how
- Part of project documentation
- Advantage in competence over competitors
- Project insurance at acquisition/development phase
- Proofing of new concepts on development phase
- Error analysing in mass production
- Saving manufacture costs by avoiding errors

The personal education in mechatronics area is not a short-period task and many as human so also hardware resources should be involved in it. With the help of virtual system, the learning personal can be quicker and cheaper to reach a higher professional state, which is a necessary condition for successful business today.

With permanent personal education, the company know-how is transferred from experts to younger generation and this know-how should be kept, also taking

into account fast changing market situation and development state in the world independent from human resources form.

Every effect modelled in the system should be remained in data base of company knowledge independent from human presence. It does not reduce the role of human factor but makes insurance factor higher.

Also corresponding documentation process belongs to consequent project loop. In documentation process, it is important to have some possibility to recover simulation state and results for possible analysis in case of manufacture problems

With rising globalisation in the world, the number of manufactures grows rapidly, almost every month, and the problem to choose the right component supplier becomes of great meaning. One of factors to distinguish among huge proposals on global commercial market is competence degree of the component manufacture. And from this point, a good simulation model is yet ‘another brick in the wall’.

It is not to wonder that wishes are rising quicker than performance. The customer should be consulted not only emotionally (‘we can do it’) but rather technically (let us analyse the simulation results). Sometimes, the customer requirements are contradictory but without technical analysis it is invisible. The system analysis tool is a good helper in this case to avoid excessive discussion.

Innovation by development is a locomotive of the progress but only flawless development and production lead to commercial success and sure business. To avoid errors by new development is one of the main tasks of every company. The virtual development is one of many steps in this trend. That is why, there are so many simulation concepts proved at present. BroSAnT gives the possibility to prove the technical innovations at early development phase without big expensive or with relative low costs.

Every error in development brings additional costs. However, it is impossible at all to develop without some errors which occur sometimes at different phases of manufacture: it can be connected with some changing of technological process or supplier. Some improvements in the manufacturing tool can have side effects which are not taken into account and so on. Any simulation tool can not find the reason itself but rather it can be served as assistant instrument to indentify cause for this failure. It should be clear, that any tool is only facility but rather ‘wonder solution’ for all problems. The quali-

ty of using tool is the same important as the professionalism of the personal working on it and tight cooperation between simulation and domain experts.

It is very important to use simulation analyses at right development phases. The simulation costs also money and time, which means in the end also money.

It is well known that the costs of error correction being close to start of production rise almost exponentially. That is why the failure analysis (failure prediction) at early development or application phases is almost obligatory for every commercial project.

Last but not least are the expenses for simulation tool. To be profitable, the simulation tool should be used as extensive as possible, which is reachable either with many experts, constantly using this tool, or with organization of some calculation task pool for it. The software manufactures are trying to sell as many software as possible, promising to increase the quality of development, but meanwhile they are silent about expanded costs for personal educational, maintenance and so on. The complexity of tools is enhanced and the average efficiency of using the tools, without correct usage, is diminished.

## 4 BroSAnT Concept

From reason above mentioned, the server-client concept is ideally appropriate for realization of system simulation. To give domain experts some possibility to use the simulation tool, it is necessary to hide complexities of simulation and modelling techniques from them. The client part should be independent, as possible as it can, from simulation environment. The focus of analysis should be laid on domain problem but rather simulation nuances.

According to this concept, the grey boxes do not have any environment of simulator and almost do not connect with peculiarities of some simulation knowledge. During the task preparing for simulation, technical knowledge of simulation object is desirable. The default setting for component is suitable for reasonable functionality of basic model. The parameter changing requires the exact understanding about what will be done. The most investigations are done in the time range because the dynamical behaviour is the focus of analysing typical mechatronics system.

The client part (Figure 1) is written in Tcl/Tk [4] and principally consists of two parts: results analyser and simulation configurator.

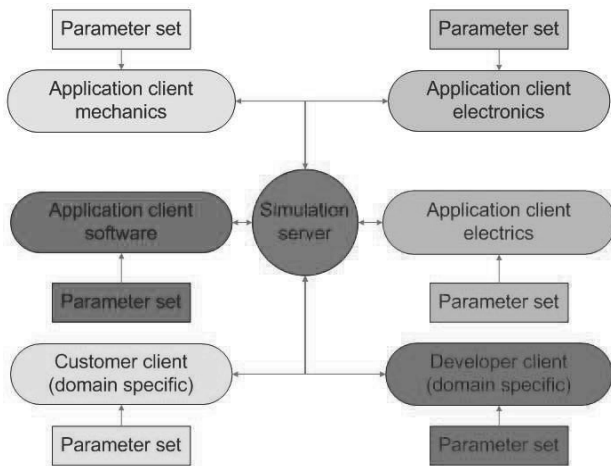


Figure 1.: Simulation concept of BroSAnT.

The first part presents ordinary two-dimension curve browser with zoom and measure facilities. The second part is responsible for simulation content: from choosing model of whole mechatronics system up to changing component parameters and necessary settings for output. At the beginning of building simulation, the macro model (device) for simulation should be chosen the first of all. After this step, the user has possibility to build the selected system from different components available in data base.

The main focus of application is modelling electrical power window regulator. It is classical example for modelling mechatronics system, because the window regulator possesses all components that are needed for mechatronics system: electronics, drive, mechanics and control part.

According to Albert Einstein ‘... we should make things as simple as possible, but not simpler’, there are controlling module and also power source for it. In controlling part, the switching sequence as time or target function should be given.

There are also a number of simulation protocols for tracking the simulation runs. The big amount of simulation possibilities should be cleared through analysing history of the parameter changing. The possibility to save user-tuned component model with help of parameter is realized by means of user library concept. All date base and personal library be kept at server space and can be use from any place if the access to calculation server is granted.

On the toolbar (Figure 2) beside zoom/ measure and different log elements, there are two important managers: simulation manager and file manager, which allow users to work with simulation results and also with measured data.

The server part is implemented in Linux, and it is responsible for simulation runs/queues and result return. To server task belongs also the service for component data base, model administration and the protocol of whole simulation processes. Thanks to the server-client solution, calculation duration is independent from client side. All calculations are performed at server and the single restriction on calculation time is the amount of required results. Practically, simulation duration is comparable with ‘real time’. Of course, it depends on simulated model and required precision.

At present the whole modelling is implemented in MAST [5] modelling language and is planed to transfer on VHDL-AMS language. With it, higher degree of freedom in selecting simulator manufacture can be reached. The simulator developer does not really support such simulation concepts, because today almost all simulators are domain oriented. Hence, there are two schema of using the simulation methods: the domain specialist makes the simulation itself based on own demand (as a rule in electronics and in algorithm development) or simulation results are produced by simulation experts according to demands from domain specialists (usually other domains). At present, Saber simulator from company Synopsys is used as simulation engine.

## 5 BroSAnT Simulation Example - Electrical Power Window Regulator

By choosing a power electrical window regulator as mechatronics system, the simulation configurator gives all necessary components to combine the whole system: different drives with sensors, mechanics on the cable basis or cross arm construction, electronics part with switches, corresponding sets of possible software algorithms, carrying door model with anti trap measure gauge, source of power like car battery and necessary controlling switch sequences. At the Figure 3 this part of configurator can be seen and one component has its open parameter set. It is drive consisting of DCPM (Direct Current Permanent Magnet) motor and worm gear.



Figure 2. Toolbar of BroSAnT.

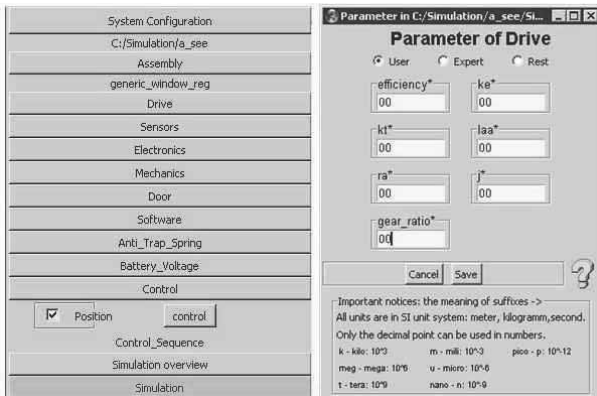


Figure 3. Component's part of the BroSAnT configurator.

After simulation run the results is delivered from server to client and the Figure 4 shows the position (in electronics unit) at left axis and maximal reached anti trap force for this system under such condition. There are four anti trapping cases and it can be seen the four different value of anti trap force before the window glass reverses (right axis). It is connected with different supplying voltage by anti trapping (car board voltage - 10 12 14 16 Volt). The supplied voltage to the motor is shown as dashed curve in conventional units on right axis. From this curve it can be concluded that voltage compensation module in algorithms is switched off. The simulation results reflect the behaviour of the system in reality. Such tests can be made with the real hardware too but sometimes it is cheaper and quicker to prove ideas virtually, especially, if hardware does not exist yet.

If the system is combined, there are many possibilities to test it from various domains. An expert from a domain can study what consequences can be happened under one or another changing in his subsystem/component or how stable his component remains under different working condition. With practical experiences using **BroSAnT**, it can be stated that different domains have each own imaginations about what is the most important in system and how it should be studied [6]. A number of practical rules can be concluded:

- Simulation results without exact information, about under which condition simulation was done, are not only useless but rather harmful
- the same with validity of test data, it is only reasonable to analyse measured data if the reliability more or less was confirmed
- the most useful information is what is known from daily business and on the contrary, what is not known is no more than interesting but not especially useful

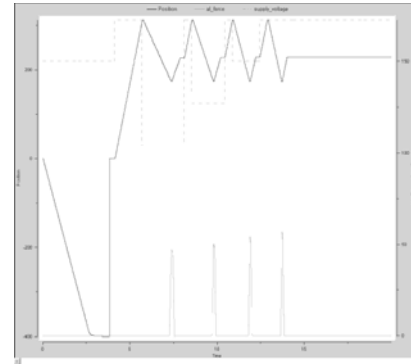


Figure 4. Result of the window regulator working against obstacle (10N/mm).

Also after cooperative working, the system simulation developer and product management, a couple 'important' remarks come in sight:

- attention to possible problems rises exponentially with rising number of failures in the field or by mass production
- avoided errors and failures will not be taken into account by profit calculation
- every end user wishes each own interface

**Acknowledgements.** We would like to thank Joachim Haase, Roland Kalb, and Detlef Ruß for their valuable feedback and advices. My great acknowledge to George A. Howlett and ActiveState team for their excellent package for 2D graphics - BLT /RBC

#### Literature

- [1] Modelica Association. (2010). *A Unified Object-Oriented Language for Physical System Modelling: Language Specification Version 3.2*. March 2010. URL: <https://www.modelica.org/documents/ModelicaSpec32.pdf>.
- [2] IEC Std61691-6/IEEE Std 1076.1. (2009). *Behavioural languages - Part 6: VHDL Analog and Mixed-Signal Extensions*. URL: [ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5465882&isnumber=5465881](http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5465882&isnumber=5465881).
- [3] SAE J2546\_200202. (2002). *Model Specification Process Standard*. SAE Electronic Design Automation Standards Committee, February 15, 2002. URL: [http://standards.sae.org/j2546\\_200202](http://standards.sae.org/j2546_200202)
- [4] ActiveTclHelp8.5.chm
- [5] Synopsis. *MAST - The Analog, Mixed-Technology and Mixed-Signal HDL for Saber*. URL: [www.synopsys.com/systems/saber/pages/mast.aspx](http://www.synopsys.com/systems/saber/pages/mast.aspx).
- [6] Petkun, S. (2011). Mechatronics simulation on system level: BroSAnT approach. *Proceeding ASIM 2011, Winterthur*, Switzerland, 7-9.09.2011.

Submitted: ASIM Winterthur June 2011

Accepted ASIM Winterthur, September 2011

Accepted SNE: March, 2012



# Training Method of Oil and Gas Industry Operators on the Example of the Automaton Simulation Usage

Eugeniy Gromakov<sup>1\*</sup>, Maria Sotnikova<sup>2</sup>

<sup>1</sup> National Research Tomsk Polytechnic University, Russia; \*[gromakov@tpu.su](mailto:gromakov@tpu.su)

<sup>2</sup> Hochschule Magdeburg-Stendal, Magdeburg, Germany

**Abstract.** At the present time, in the oil and gas industries centers of operators training simulate situational problems like alarm and pre-alarm events. Recent studies in this field allow us to consider only the most likely emergency scenarios. In order to identify unlikely and untypical situations the HAZOP-methodology is being used. Its usage is becoming problematic due to the significant growth of low-probability *fault tree eaves*.

In this paper we propose to use the automaton approach with the use of advanced automaton cuts for the effective use of failures models in practice, training of operators working in emergency situations. This process simulation approach will allow taking into account highly unlikely and untypical situations.

## Introduction

Accidents at various facilities in the oil and gas sector can be divided into three groups of interrelated causes that contribute to the emergence of the accident, as analyses have shown:

- equipment failures,
- human error,
- underexpanded external effects of natural and manmade.

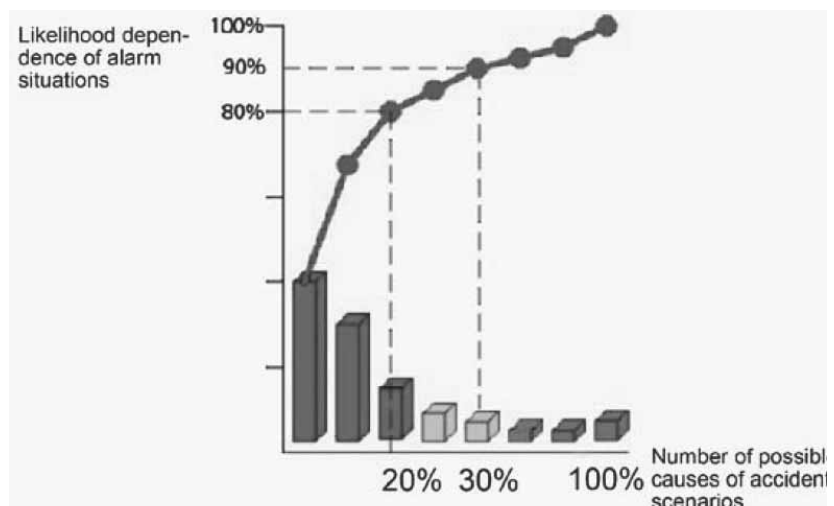
Through training and recurrent training of operators, we can reduce the severity of the accidents. At the present time, in industrial centers of Operators training situational problems like typical alarm and pre-alarm events are considered.

## 1 Formulation of the Problem

According to the Pareto analysis only 20% of all incidents are responsible for 80% of damages. As situational problems 20% of the best reasons of accident scenarios are detected and considered. In this case consideration of these situational problems reduces the likelihood of emergency events up to 80%.

Thus it turns out that 80% of the scenarios fall out of consideration in the virtual training places, and operators are not prepared for 80% of the unlikely system failures causes. However, international experience indicates greater risk of unlikely and untypical situations which operators are not prepared to control.

Emergency situation in the oil industry can paralyze the work of the oil and gas facilities until the elimination of the causes, and if unsuccessful can result in fire, possible detonation, and others.



**Figure 1.** Likelihood of alarm situation, depending on the number of possible causes of the accident scenarios.

That may lead to the death of more than one dozen people. Recent studies of the fault tree construction allow us to consider only the most likely emergency scenarios. That makes it necessary to develop the methods to detect more untypical events. It will allow operators to be ready to manage a lot of emergency events. Thereby it will significantly improve the quality of their training.

In Figure 1. there is the likelihood dependence of emergency situations, depending on the number of possible causes of the accident scenarios.

## 2 The Method of Finite Automaton as a Tool for Studying of a Typical Emergency Scenario

### 2.1 Disadvantages of the HAZOP-Methodology used to Identify Emergency Causes

In order to identify emergency causes at the present time the HAZOP-methodology is used. [1].

But the HAZOP-methodology considers only the most dangerous and most likely emergency situations. Its usage is becoming problematic to identify unlikely and untypical situations due to the significant growth of low-probability 'a fault tree leaves'. In addition, the HAZOP methodology has a number of other shortcomings. In the process of the object analyzing on the availability of emergency situations consideration of relationships, regularities of accident scenarios, co-occurrence of multiple emergency scenarios are not included. [2].

### 2.2 Usage of the Extended Automaton Cuts

In the practice of operators training of working in emergency situations automaton approach is proposed using 'cuts' of 'the extended machine'. This approach will reduce costs for expensive prevention of emergency situations. Such an automaton approach allows creating virtual field automated place.

That allows regularly training of operators for emergency scenarios and also gives intelligence prompts information of appropriate action.

This approach has considerable practical importance. Enterprises can get the simulation methods of technological processes to the workplace of the operators that will allow taking into account not unlikely typical situations. This account will allow a more carefully prepare operators of the oil and gas industry. This in turn will significantly reduce the probability of accidents and prevent them. That will reduce the costs of emergency response and resulting consequences.

It becomes cost effective to prepare the operators to the emergence of a larger number of alarm events, than to technically realize the lack of all contingencies. StateFlow is used as software package for automated device modeling. StateFlow is a simulation tool for event-driven systems, for designing of complex control systems.

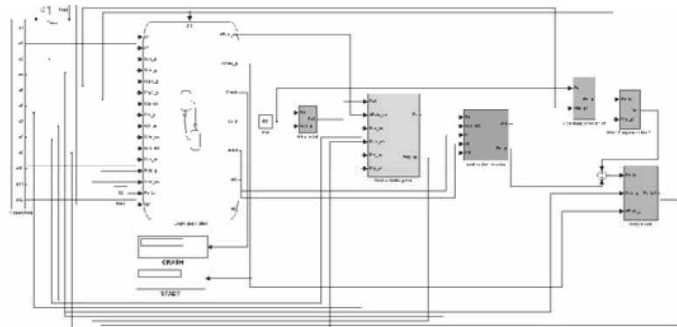


Figure 2. Finite state machine model of technological process section.



Figure 3. StateFlow-state model of the technological process



## 4 Conclusion

At the present stage of the oil and gas operators' training, carried out in training centers, only typical emergency situations are considered. The importance of playing out such scenarios is significant, but we thus disregard most of the causes of other accident scenarios. The usage of "advanced finite automaton" will help us to consider significantly more events. We will limit the amount of the low-robability "leaves" of the "fault tree" by the "cuts" usage of "extended automata".

Field automated virtual places will not only generate motor-reflex skills of action in dangerous situations, but also will demonstrate to the operators the processes taking place in the equipment during the technological processes and their interdependence.

## References

- [1] IEC 61882:2001. (2003). *Hazard and operability studies (HAZOP studies) – Application guide*, British standard.
- [2] AER-5437 TSI 41-018: *Guidelines for conducting HAZOP studies*, SAUDI ARAMCO, loss prevention department.

Submitted: ASIM Winterthur June 2011

Accepted ASIM Winterthur, September 2011

Accepted SNE Short Note: March, 2012

# SNE Simulation News

## EUROSIM Data and Quick Info



## EUROSIM 2013

### 8<sup>th</sup> EUROSIM Congress on Modelling and Simulation

The City Hall, Cardiff, Wales, United Kingdom 10-13 September 2013

[www.eurosim2013.info](http://www.eurosim2013.info)

#### Contents

Info EUROSIM .....	2
Info EUROSIM Societies .....	3 - 7
Info ASIM, CROSSIM .....	3
Info CSSS, HSS, DBSS, FRANCOSIM .....	4
Info ISCS, PSCS, SIMS, SLOSIM .....	5
Info UKSIM, LSS, CAE-SMSG, ROMSIM .....	6
Info RNSS, LIOPHANT Info SNE .....	7
News EUROSIM, ASIM, SLOSIM .....	9
Introduction & News RNSS – Russian Sim. Society .....	10

**Simulation Notes Europe** SNE is the official membership journal of EUROSIM and distributed / available to members of the EUROSIM Societies as part of the membership benefits. SNE is published in a printed version (Print ISSN 2305-9974) and in an online version (Online ISSN 2306-0271). With Online SNE the publisher ARGESIM follows the Open Access strategy for basic SNE contributions. Since 2011 Online SNE contributions are identified by DOI 10.11128/sne.xx.nnnnn. for better web availability and indexing.

Print SNE, high-resolution Online SNE, and additional SNE contributions are available for subscription via membership in a EUROSIM society.

This *EUROSIM Data & Quick Info* compiles data from EUROSIM societies and groups: addresses, weblinks, officers of societies with function and email, to be published regularly in SNE issues.

#### SNE Reports Editorial Board

EUROSIM Khalid Al-Begain, [kbegin@glam.ac.uk](mailto:kbegin@glam.ac.uk)  
Borut Zupančič, [borut.zupancic@fe.uni-lj.si](mailto:borut.zupancic@fe.uni-lj.si)  
Felix Breitenecker, [Felix.Breitenecker@tuwien.ac.at](mailto:Felix.Breitenecker@tuwien.ac.at)  
ASIM Thorsten Pawletta, [pawel@mb.hs-wismar.de](mailto:pawel@mb.hs-wismar.de)  
CROSSIM Vesna Dušak, [vdusak@foi.hr](mailto:vdusak@foi.hr)  
CSSS Mikuláš Alexík, [alexik@frtk.utc.sk](mailto:alexik@frtk.utc.sk)  
DBSS A. Heemink, [a.w.heemink@its.tudelft.nl](mailto:a.w.heemink@its.tudelft.nl)  
FRANCOSIM Karim Djouani, [djouani@u-pec.fr](mailto:djouani@u-pec.fr)  
HSS András Jávör, [javor@eik.bme.hu](mailto:javor@eik.bme.hu)  
ISCS M. Savastano, [mario.savastano@unina.it](mailto:mario.savastano@unina.it)  
PSCS Zenon Sosnowski, [zenon@ii.pb.bialystok.pl](mailto:zenon@ii.pb.bialystok.pl)  
SIMS Esko Juuso, [esko.juuso@oulu.fi](mailto:esko.juuso@oulu.fi)  
SLOSIM Rihard Karba, [rihard.karba@fe.uni-lj.si](mailto:rihard.karba@fe.uni-lj.si)  
UKSIM Richard Zobel, [r.zobel@ntlworld.com](mailto:r.zobel@ntlworld.com)  
CAE-SMSG Emilio Jiminez, [emilio.jiminez@unirioja.es](mailto:emilio.jiminez@unirioja.es)  
LSS Yuri Merkuryev, [merkur@itl.rtu.lv](mailto:merkur@itl.rtu.lv)  
ROMSIM Florin Stanculescu, [sflorin@ici.ro](mailto:sflorin@ici.ro)  
RNSS Y. Senichenkov, [sneyb@dcn.infos.ru](mailto:sneyb@dcn.infos.ru)  
LIOPHANT F. Longo, [f.longo@unical.it](mailto:f.longo@unical.it)

#### SNE Editorial Office /ARGESIM

→ [www.sne-journal.org](http://www.sne-journal.org), [www.eurosim.info](http://www.eurosim.info)

✉ [office@sne-journal.org](mailto:office@sne-journal.org)

Felix Breitenecker, [eic@sne-journal.org](mailto:eic@sne-journal.org)  
Anna Mathe, [Anna.Mathe@tuwien.ac.at](mailto:Anna.Mathe@tuwien.ac.at), [office@sne-journal.org](mailto:office@sne-journal.org)  
Nikolas Popper, [Niki.Popper@drahtwarenhandlung.at](mailto:Niki.Popper@drahtwarenhandlung.at)

If you have any information, announcement, etc. you want to see published, please contact a member of the editorial board in your country or the editorial office.



## EUROSIM Federation of European Simulation Societies

**General Information.** EUROSIM, the Federation of European Simulation Societies, was set up in 1989. The purpose of EUROSIM is to provide a European forum for regional and national simulation societies to promote the advancement of modelling and simulation in industry, research, and development.

→ [www.eurosim.info](http://www.eurosim.info)

**Member Societies.** EUROSIM members may be national simulation societies and regional or international societies and groups dealing with modelling and simulation. At present EUROSIM has thirteen full members and three observer member:

ASIM	Arbeitsgemeinschaft Simulation <i>Austria, Germany, Switzerland</i>
CEA-SMSG	Spanish Modelling and Simulation Group <i>Spain</i>
CROSSIM	Croatian Society for Simulation Modeling <i>Croatia</i>
CSSS	Czech and Slovak Simulation Society <i>Czech Republic, Slovak Republic</i>
DBSS	Dutch Benelux Simulation Society <i>Belgium, Netherlands</i>
FRANCO-SIM	Société Francophone de Simulation <i>Belgium, France</i>
HSS	Hungarian Simulation Society <i>Hungary</i>
ISCS	Italian Society for Computer Simulation <i>Italy</i>
LSS	Latvian Simulation Society <i>Latvia</i>
PSCS	Polish Society for Computer Simulation <i>Poland</i>
SIMS	Simulation Society of Scandinavia <i>Denmark, Finland, Norway, Sweden</i>
SLOSIM	Slovenian Simulation Society <i>Slovenia</i>
UKSIM	United Kingdom Simulation Society <i>UK, Ireland</i>
ROMSIM	Romanian Society for Modelling and Simulation, <i>Romania, Observer Member</i>
RNSS	Russian National Simulation Society <i>Russian Federation, Observer Member</i>
LIOPHANT	LIOPHANT Simulation Club <i>Italy &amp; International, Observer Member</i>

EUROSIM Board / Officers. EUROSIM is governed by a board consisting of one representative of each member society, president and past president, and representatives for SNE Simulation notes Europe. The President is nominated by the society organising the next EUROSIM Congress. Secretary and Treasurer are elected out of members of the Board.

President	Khalid Al.Begain <i>kbegain@glam.ac.uk</i>
Past President	Mikuláš Alexík (CSSS), <i>alexik@frtk.fri.utc.sk</i>
Secretary	Borut Zupančič (SLOSIM) <i>borut.zupancic@fe.uni-lj.si</i>
Treasurer	Felix Breitenecker (ASIM) <i>felix.breitenecker@tuwien.ac.at</i>
SNE Repres.	Felix Breitenecker <i>felix.breitenecker@tuwien.ac.at</i>

SNE – Simulation Notes Europe. SNE is a scientific journal with reviewed contributions as well as a membership newsletter for EUROSIM with information from the societies in the *News Section*. EUROSIM societies are offered to distribute to their members the journal SNE as official membership journal. SNE Publishers are EUROSIM, ARGESIM and ASIM.

Editor-in-chief	Felix Breitenecker <i>felix.breitenecker@tuwien.ac.at</i>
-----------------	--

→ [www.sne-journal.org](http://www.sne-journal.org),

✉ [office@sne-journal.org](mailto:office@sne-journal.org)

EUROSIM Congress. EUROSIM is running the triennial conference series EUROSIM Congress. The congress is organised by one of the EUROSIM societies.

EUROSIM 2013 will be organised by UKSIM in Cardiff, Wales, UK, September 10-13, 2013.

### Chairs / Team EUROSIM 2013

Khalid Al.Begain, *kbegain@glam.ac.uk*  
Richard Zobel, *r.zobel@ntlworld.com*  
David Al-Dabass, *david.al-dabass@ntu.ac.uk*  
Alessandra Orsoni, *a.orsoni@kingston.ac.uk*  
Richard Cant, *richard.cant@ntu.ac.uk*

→ [www.eurosim2013.info](http://www.eurosim2013.info)



## ASIM German Simulation Society Arbeitsgemeinschaft Simulation

ASIM (Arbeitsgemeinschaft Simulation) is the association for simulation in the German speaking area, servicing mainly Germany, Switzerland and Austria. ASIM was founded in 1981 and has now about 700 individual members, and 30 institutional or industrial members. Furthermore, ASIM counts about 300 affiliated members.

→ [www.asim-gi.org](http://www.asim-gi.org) with members' area

✉ [info@asim-gi.org](mailto:info@asim-gi.org), [admin@asim-gi.org](mailto:admin@asim-gi.org)

✉ ASIM – Inst. f. Analysis and Scientific Computing  
Vienna University of Technology  
Wiedner Hauptstraße 8-10, 1040 Vienna, Austria

### ASIM Officers

<b>President</b>	Felix Breitenecker <a href="mailto:felix.breitenecker@tuwien.ac.at">felix.breitenecker@tuwien.ac.at</a>
<b>Vice presidents</b>	Sigrid Wenzel, <a href="mailto:s.wenzel@uni-kassel.de">s.wenzel@uni-kassel.de</a> T. Pawletta, <a href="mailto:pawel@mb.hs-wismar.de">pawel@mb.hs-wismar.de</a>
<b>Secretary</b>	Anna Mathe, <a href="mailto:anna.mathe@tuwien.ac.at">anna.mathe@tuwien.ac.at</a>
<b>Treasurer</b>	I. Bausch-Gall, <a href="mailto:Ingrid@Bausch-Gall.de">Ingrid@Bausch-Gall.de</a>
<b>Membership Affairs</b>	S. Wenzel, <a href="mailto:s.wenzel@uni-kassel.de">s.wenzel@uni-kassel.de</a> W. Maurer, <a href="mailto:werner.maurer@zhwin.ch">werner.maurer@zhwin.ch</a> Ch. Deatcu, <a href="mailto:christina.deatcu@hs-wismar.de">christina.deatcu@hs-wismar.de</a> F. Breitenecker, <a href="mailto:felix.breitenecker@tuwien.ac.at">felix.breitenecker@tuwien.ac.at</a>
<b>Universities / Research Inst.</b>	S. Wenzel, <a href="mailto:s.wenzel@uni-kassel.de">s.wenzel@uni-kassel.de</a> W. Wiechert, <a href="mailto:W.Wiechert@fz-juelich.de">W.Wiechert@fz-juelich.de</a> J. Haase, <a href="mailto:Joachim.Haase@eas.iis.fraunhofer.de">Joachim.Haase@eas.iis.fraunhofer.de</a> Katharina Nöh, <a href="mailto:k.noeh@fz-juelich.de">k.noeh@fz-juelich.de</a>
<b>Industry</b>	S. Wenzel, <a href="mailto:s.wenzel@uni-kassel.de">s.wenzel@uni-kassel.de</a> K. Panreck, <a href="mailto:Klaus.Panreck@hella.com">Klaus.Panreck@hella.com</a>
<b>Conferences</b>	Klaus Panreck <a href="mailto:Klaus.Panreck@hella.com">Klaus.Panreck@hella.com</a> A. Gnauck, <a href="mailto:albrecht.gnauck@tu-cottbus.de">albrecht.gnauck@tu-cottbus.de</a>
<b>Publications</b>	Th. Pawletta, <a href="mailto:pawel@mb.hs-wismar.de">pawel@mb.hs-wismar.de</a> Christina Deatcu, <a href="mailto:christina.deatcu@hs-wismar.de">christina.deatcu@hs-wismar.de</a> F. Breitenecker, <a href="mailto:felix.breitenecker@tuwien.ac.at">felix.breitenecker@tuwien.ac.at</a>
<b>Repr. EuroSIM</b>	F. Breitenecker, <a href="mailto:felix.breitenecker@tuwien.ac.at">felix.breitenecker@tuwien.ac.at</a> N. Popper, <a href="mailto:niki.popper@drahtwarenhandlung.at">niki.popper@drahtwarenhandlung.at</a>
<b>Education / Teaching</b>	Ch. Deatcu, <a href="mailto:christina.deatcu@hs-wismar.de">christina.deatcu@hs-wismar.de</a> N. Popper, <a href="mailto:niki.popper@drahtwarenhandlung.at">niki.popper@drahtwarenhandlung.at</a> Katharina Nöh, <a href="mailto:k.noeh@fz-juelich.de">k.noeh@fz-juelich.de</a>
<b>International Affairs</b>	H. Szczerbicka, <a href="mailto:hsz@sim.uni-hannover.de">hsz@sim.uni-hannover.de</a> O. Rose, <a href="mailto:Oliver.Rose@tu-dresden.de">Oliver.Rose@tu-dresden.de</a>
<b>Editorial Board SNE</b>	T. Pawletta, <a href="mailto:pawel@mb.hs-wismar.de">pawel@mb.hs-wismar.de</a> Ch. Deatcu, <a href="mailto:christina.deatcu@hs-wismar.de">christina.deatcu@hs-wismar.de</a>
<b>Web EuroSIM</b>	Anna Mathe, <a href="mailto:anna.mathe@tuwien.ac.at">anna.mathe@tuwien.ac.at</a>

Last data update December 2012

ASIM Working Committee. ASIM, part of GI - Gesellschaft für Informatik, is organised in Working Committees, dealing with applications and comprehensive subjects in modelling and simulation:

### ASIM Working Committee

<b>GMMS</b>	Methods in Modelling and Simulation Th. Pawletta, <a href="mailto:pawel@mb.hs-wismar.de">pawel@mb.hs-wismar.de</a>
<b>SUG</b>	Simulation in Environmental Systems Wittmann, <a href="mailto:wittmann@informatik.uni-hamburg.de">wittmann@informatik.uni-hamburg.de</a>
<b>STS</b>	Simulation of Technical Systems H.T.Mammen, <a href="mailto:Heinz-Theo.Mammen@hella.com">Heinz-Theo.Mammen@hella.com</a>
<b>SPL</b>	Simulation in Production and Logistics Sigrid Wenzel, <a href="mailto:s.wenzel@uni-kassel.de">s.wenzel@uni-kassel.de</a>
<b>EDU</b>	Simulation in Education/Education in Simulation N. Popper, <a href="mailto:niki.popper@drahtwarenhandlung.at">niki.popper@drahtwarenhandlung.at</a>

Working Groups for Simulation in Business Administration, in Traffic Systems, for Standardisation, for Validation, etc.

## CROSSIM – Croatian Society for Simulation Modelling

CROSSIM-Croatian Society for Simulation Modelling was founded in 1992 as a non-profit society with the goal to promote knowledge and use of simulation methods and techniques and development of education. CROSSIM is a full member of EUROSIM since 1997.

→ [www.eurosim.info](http://www.eurosim.info)

✉ [vdusak@foi.hr](mailto:vdusak@foi.hr)

✉ CROSSIM / Vesna Dušak  
Faculty of Organization and Informatics Varaždin, University of Zagreb  
Pavlinska 2, HR-42000 Varaždin, Croatia

### CROSSIM Officers

<b>President</b>	Vesna Dušak, <a href="mailto:vdusak@foi.hr">vdusak@foi.hr</a>
<b>Vice president</b>	Jadranka Božikov, <a href="mailto:jbozikov@snz.hr">jbozikov@snz.hr</a>
<b>Secretary</b>	Vesna Bosilj-Vukšić, <a href="mailto:vbosilj@efzg.hr">vbosilj@efzg.hr</a>
<b>Executive board members</b>	Vlatko Čerić, <a href="mailto:vceric@efzg.hr">vceric@efzg.hr</a> Tarzan Legović, <a href="mailto:legovic@irb.hr">legovic@irb.hr</a>
<b>Repr. EuroSIM</b>	Jadranka Božikov, <a href="mailto:jbozikov@snz.hr">jbozikov@snz.hr</a>
<b>Edit. Board SNE</b>	Vesna Dušak, <a href="mailto:vdusak@foi.hr">vdusak@foi.hr</a>
<b>Web EuroSIM</b>	Jadranka Božikov, <a href="mailto:jbozikov@snz.hr">jbozikov@snz.hr</a>

Last data update December 2012



## CSSS – Czech and Slovak Simulation Society

CSSS -The *Czech and Slovak Simulation Society* has about 150 members working in Czech and Slovak national scientific and technical societies (*Czech Society for Applied Cybernetics and Informatics, Slovak Society for Applied Cybernetics and Informatics*). The main objectives of the society are: development of education and training in the field of modelling and simulation, organising professional workshops and conferences, disseminating information about modelling and simulation activities in Europe. Since 1992, CSSS is full member of EUROSIM.

→ [www.fit.vutbr.cz/CSSS](http://www.fit.vutbr.cz/CSSS)

✉ [snorek@fel.cvut.cz](mailto:snorek@fel.cvut.cz)

✉ CSSS / Miroslav Šnorek, CTU Prague  
FEE, Dept. Computer Science and Engineering,  
Karlovo nám. 13, 121 35 Praha 2, Czech Republic

### CSSS Officers

<b>President</b>	Miroslav Šnorek, <a href="mailto:snorek@fel.cvut.cz">snorek@fel.cvut.cz</a>
<b>Vice president</b>	Mikuláš Alexik, <a href="mailto:alexik@frtk.fri.utc.sk">alexik@frtk.fri.utc.sk</a>
<b>Treasurer</b>	Evžen Kindler, <a href="mailto:ekindler@centrum.cz">ekindler@centrum.cz</a>
<b>Scientific Secr.</b>	A. Kavička, <a href="mailto:Antonin.Kavicka@upce.cz">Antonin.Kavicka@upce.cz</a>
<b>Repr. EUROSIM</b>	Miroslav Šnorek, <a href="mailto:snorek@fel.cvut.cz">snorek@fel.cvut.cz</a>
<b>Deputy</b>	Mikuláš Alexik, <a href="mailto:alexik@frtk.fri.utc.sk">alexik@frtk.fri.utc.sk</a>
<b>Edit. Board SNE</b>	Mikuláš Alexik, <a href="mailto:alexik@frtk.fri.utc.sk">alexik@frtk.fri.utc.sk</a>
<b>Web EUROSIM</b>	Petr Peringer, <a href="mailto:peringer@fit.vutbr.cz">peringer@fit.vutbr.cz</a>

*Last data update December 2012*

## FRANCOSIM – Société Francophone de Simulation

FRANCOSIM was founded in 1991 and aims to the promotion of simulation and research, in industry and academic fields. Francosim operates two poles.

- Pole Modelling and simulation of discrete event systems. Pole Contact: *Henri Pierreval*, [pierre-va@imfa.fr](mailto:pierre-va@imfa.fr)
- Pole Modelling and simulation of continuous systems. Pole Contact: *Yskandar Hamam*, [y.hamam@esiee.fr](mailto:y.hamam@esiee.fr)

→ [www.eurosim.info](http://www.eurosim.info)

✉ [y.hamam@esiee.fr](mailto:y.hamam@esiee.fr)

✉ FRANCOSIM / Yskandar Hamam  
Groupe ESIEE, Cité Descartes,  
BP 99, 2 Bd. Blaise Pascal,  
93162 Noisy le Grand CEDEX, France

### FRANCOSIM Officers

<b>President</b>	Karim Djouani, <a href="mailto:djouani@u-pec.fr">djouani@u-pec.fr</a>
<b>Treasurer</b>	François Rocaries, <a href="mailto:f.rocaries@esiee.fr">f.rocaries@esiee.fr</a>
<b>Repr. EUROSIM</b>	Karim Djouani, <a href="mailto:djouani@u-pec.fr">djouani@u-pec.fr</a>
<b>Edit. Board SNE</b>	Karim Djouani, <a href="mailto:djouani@u-pec.fr">djouani@u-pec.fr</a>

*Last data update December 2012*

## DBSS – Dutch Benelux Simulation Society

The Dutch Benelux Simulation Society (DBSS) was founded in July 1986 in order to create an organisation of simulation professionals within the Dutch language area. DBSS has actively promoted creation of similar organisations in other language areas. DBSS is a member of EUROSIM and works in close cooperation with its members and with affiliated societies.

→ [www.eurosim.info](http://www.eurosim.info)

✉ [a.w.heemink@its.tudelft.nl](mailto:a.w.heemink@its.tudelft.nl)

✉ DBSS / A. W. Heemink  
Delft University of Technology, ITS - twi,  
Mekelweg 4, 2628 CD Delft, The Netherlands

### DBSS Officers

<b>President</b>	A. Heemink, <a href="mailto:a.w.heemink@its.tudelft.nl">a.w.heemink@its.tudelft.nl</a>
<b>Vice president</b>	W. Smit, <a href="mailto:smitnet@wxs.nl">smitnet@wxs.nl</a>
<b>Treasurer</b>	W. Smit, <a href="mailto:smitnet@wxs.nl">smitnet@wxs.nl</a>
<b>Secretary</b>	W. Smit, <a href="mailto:smitnet@wxs.nl">smitnet@wxs.nl</a>
<b>Repr. EUROSIM</b>	A. Heemink, <a href="mailto:a.w.heemink@its.tudelft.nl">a.w.heemink@its.tudelft.nl</a>
<b>Deputy</b>	W. Smit, <a href="mailto:smitnet@wxs.nl">smitnet@wxs.nl</a>
<b>Edit. Board SNE</b>	A. Heemink, <a href="mailto:a.w.heemink@its.tudelft.nl">a.w.heemink@its.tudelft.nl</a>

*Last data update April 2006*

## HSS – Hungarian Simulation Society

The Hungarian Member Society of EUROSIM was established in 1981 as an association promoting the exchange of information within the community of people involved in research, development, application and education of simulation in Hungary and also contributing to the enhancement of exchanging information between the Hungarian simulation community and the simulation communities abroad. HSS deals with the organization of lectures, exhibitions, demonstrations, and conferences.

→ [www.eurosim.info](http://www.eurosim.info)

✉ [javor@eik.bme.hu](mailto:javor@eik.bme.hu)

✉ HSS / András Jávör,  
Budapest Univ. of Technology and Economics,  
Sztoczek u. 4, 1111 Budapest, Hungary



**HSS Officers**

<b>President</b>	András Jávör, <a href="mailto:javor@eik.bme.hu">javor@eik.bme.hu</a>
<b>Vice president</b>	Gábor Szűcs, <a href="mailto:szucs@itm.bme.hu">szucs@itm.bme.hu</a>
<b>Secretary</b>	Ágnes Vigh, <a href="mailto:vigh@itm.bme.hu">vigh@itm.bme.hu</a>
<b>Repr. EUROSIM</b>	András Jávör, <a href="mailto:javor@eik.bme.hu">javor@eik.bme.hu</a>
<b>Deputy</b>	Gábor Szűcs, <a href="mailto:szucs@itm.bme.hu">szucs@itm.bme.hu</a>
<b>Edit. Board SNE</b>	András Jávör, <a href="mailto:javor@eik.bme.hu">javor@eik.bme.hu</a>
<b>Web EUROSIM</b>	Gábor Szűcs, <a href="mailto:szucs@itm.bme.hu">szucs@itm.bme.hu</a>

*Last data update March 2008***PSCS – Polish Society for Computer Simulation - update**

PSCS was founded in 1993 in Warsaw. PSCS is a scientific, non-profit association of members from universities, research institutes and industry in Poland with common interests in variety of methods of computer simulations and its applications. At present PSCS counts 257 members.

→ [www.ptsk.man.bialystok.pl](http://www.ptsk.man.bialystok.pl)✉ [leon@ibib.waw.pl](mailto:leon@ibib.waw.pl)✉ PSCS / Leon Bobrowski, c/o IBIB PAN,  
ul. Trojdena 4 (p.416), 02-109 Warszawa, Poland**PSCS Officers**

<b>President</b>	Leon Bobrowski, <a href="mailto:leon@ibib.waw.pl">leon@ibib.waw.pl</a>
<b>Vice president</b>	Andrzej Grzyb, Tadeusz Nowicki
<b>Treasurer</b>	Z. Sosnowski, <a href="mailto:zenon@ii.pb.bialystok.pl">zenon@ii.pb.bialystok.pl</a>
<b>Secretary</b>	Zdzisław Galkowski, <a href="mailto:Zdzislaw.Galkowski@simr.pw.edu.pl">Zdzislaw.Galkowski@simr.pw.edu.pl</a>
<b>Repr. EUROSIM</b>	Leon Bobrowski, <a href="mailto:leon@ibib.waw.pl">leon@ibib.waw.pl</a>
<b>Deputy</b>	Tadeusz Nowicki, <a href="mailto:tadeusz.nowicki@wat.edu.pl">tadeusz.nowicki@wat.edu.pl</a>
<b>Edit. Board SNE</b>	Zenon Sosnowski, <a href="mailto:z.sosnowski@pb.edu.pl">z.sosnowski@pb.edu.pl</a>
<b>Web EUROSIM</b>	Magdalena Topczewska <a href="mailto:m.topczewska@pb.edu.pl">m.topczewska@pb.edu.pl</a>

*Last data update December 2012***ISCS – Italian Society for Computer Simulation**

The Italian Society for Computer Simulation (ISCS) is a scientific non-profit association of members from industry, university, education and several public and research institutions with common interest in all fields of computer simulation.

→ [www.eurosim.info](http://www.eurosim.info)✉ [Mario.savastano@uniina.it](mailto:Mario.savastano@uniina.it)✉ ISCS / Mario Savastano,  
c/o CNR - IRSIP,  
Via Claudio 21, 80125 Napoli, Italy**ISCS Officers**

<b>President</b>	M. Savastano, <a href="mailto:mario.savastano@uniina.it">mario.savastano@uniina.it</a>
<b>Vice president</b>	F. Maceri, <a href="mailto:Franco.Maceri@uniroma2.it">Franco.Maceri@uniroma2.it</a>
<b>Repr. EUROSIM</b>	F. Maceri, <a href="mailto:Franco.Maceri@uniroma2.it">Franco.Maceri@uniroma2.it</a>
<b>Secretary</b>	Paola Provenzano, <a href="mailto:paola.provenzano@uniroma2.it">paola.provenzano@uniroma2.it</a>
<b>Edit. Board SNE</b>	M. Savastano, <a href="mailto:mario.savastano@uniina.it">mario.savastano@uniina.it</a>

*Last data update December 2012***SIMS – Scandinavian Simulation Society**

SIMS is the *Scandinavian Simulation Society* with members from the four Nordic countries Denmark, Finland, Norway and Sweden. The SIMS history goes back to 1959. SIMS practical matters are taken care of by the SIMS board consisting of two representatives from each Nordic country (Iceland one board member).

**SIMS Structure.** SIMS is organised as federation of regional societies. There are FinSim (Finnish Simulation Forum), DKSIM (Dansk Simuleringsforening) and NFA (Norsk Forening for Automatisering).

→ [www.scansims.org](http://www.scansims.org)✉ [esko.juuso@oulu.fi](mailto:esko.juuso@oulu.fi)

✉ SIMS / Esko Juuso, Department of Process and Environmental Engineering, 90014 Univ.Oulu, Finland

**SIMS Officers**

<b>President</b>	Esko Juuso, <a href="mailto:esko.juuso@oulu.fi">esko.juuso@oulu.fi</a>
<b>Vice president</b>	Erik Dahlquist, <a href="mailto:erik.dahlquist@mdh.se">erik.dahlquist@mdh.se</a>
<b>Treasurer</b>	Vadim Engelson, <a href="mailto:vadim.engelson@mathcore.com">vadim.engelson@mathcore.com</a>
<b>Repr. EUROSIM</b>	Esko Juuso, <a href="mailto:esko.juuso@oulu.fi">esko.juuso@oulu.fi</a>
<b>Edit. Board SNE</b>	Esko Juuso, <a href="mailto:esko.juuso@oulu.fi">esko.juuso@oulu.fi</a>
<b>Web EUROSIM</b>	Vadim Engelson

*Last data update December 2012***SLOSIM – Slovenian Society for Simulation and Modelling**

SLOSIM - Slovenian Society for Simulation and Modelling was established in 1994 and became the full member of EUROSIM in 1996. Currently it has 69



members from both slovenian universities, institutes, and industry. It promotes modelling and simulation approaches to problem solving in industrial as well as in academic environments by establishing communication and cooperation among corresponding teams.

→ [www.slosim.si](http://www.slosim.si)✉ [slosim@fe.uni-lj.si](mailto:slosim@fe.uni-lj.si)✉ SLOSIM / Rihard Karba, Faculty of Electrical Engineering, University of Ljubljana,  
Tržaška 25, 1000 Ljubljana, Slovenia



SLOSIM Officers	
<b>President</b>	B. Zupančič, <a href="mailto:borut.zupancic@fe.uni-lj.si">borut.zupancic@fe.uni-lj.si</a>
<b>Vice president</b>	Leon Žlajpah, <a href="mailto:leon.zlajpah@ijs.si">leon.zlajpah@ijs.si</a>
<b>Secretary</b>	Vito Logar, <a href="mailto:vito.logar@fe.uni-lj.si">vito.logar@fe.uni-lj.si</a>
<b>Treasurer</b>	Milan Simčič, <a href="mailto:milan.simcic@fe.uni-lj.si">milan.simcic@fe.uni-lj.si</a>
<b>Repr. EUROSIM</b>	B. Zupančič, <a href="mailto:borut.zupancic@fe.uni-lj.si">borut.zupancic@fe.uni-lj.si</a>
<b>Deputy</b>	Rihard Karba, <a href="mailto:rihard.karba@fe.uni-lj.si">rihard.karba@fe.uni-lj.si</a>
<b>Edit. Board SNE</b>	Rihard Karba, <a href="mailto:rihard.karba@fe.uni-lj.si">rihard.karba@fe.uni-lj.si</a>
<b>Web EUROSIM</b>	Vito Logar, <a href="mailto:vito.logar@fe.uni-lj.si">vito.logar@fe.uni-lj.si</a>

*Last data update December 2012*

## UKSIM - United Kingdom Simulation Society

UKSIM has more than 100 members throughout the UK from universities and industry. It is active in all areas of simulation and it holds a biennial conference as well as regular meetings and workshops.

→ [www.uksim.org.uk](http://www.uksim.org.uk)

✉ [david.al-dabass@ntu.ac.uk](mailto:david.al-dabass@ntu.ac.uk)

- ✉ UKSIM / Prof. David Al-Dabass  
Computing & Informatics,  
Nottingham Trent University  
Clifton lane, Nottingham, NG11 8NS  
United Kingdom

UKSIM Officers	
<b>President</b>	David Al-Dabass, <a href="mailto:david.al-dabass@ntu.ac.uk">david.al-dabass@ntu.ac.uk</a>
<b>Vice president</b>	A. Orsoni, <a href="mailto:A.Orsoni@kingston.ac.uk">A.Orsoni@kingston.ac.uk</a>
<b>Secretary</b>	Richard Cant, <a href="mailto:richard.cant@ntu.ac.uk">richard.cant@ntu.ac.uk</a>
<b>Treasurer</b>	A. Orsoni, <a href="mailto:A.Orsoni@kingston.ac.uk">A.Orsoni@kingston.ac.uk</a>
<b>Membership chair</b>	K. Al-Begain, <a href="mailto:kbegain@glam.ac.uk">kbegain@glam.ac.uk</a>
<b>Univ. liaison chair</b>	R. Cheng, <a href="mailto:rchc@maths.soton.ac.uk">rchc@maths.soton.ac.uk</a>
<b>Repr. EUROSIM</b>	Richard Zobel, <a href="mailto:r.zobel@ntlworld.com">r.zobel@ntlworld.com</a>
<b>Deputy</b>	K. Al-Begain, <a href="mailto:kbegain@glam.ac.uk">kbegain@glam.ac.uk</a>
<b>Edit. Board SNE</b>	Richard Zobel, <a href="mailto:r.zobel@ntlworld.com">r.zobel@ntlworld.com</a>

*Last data update December 2012*

## CEA-SMSG – Spanish Modelling and Simulation Group

CEA is the Spanish Society on Automation and Control. In order to improve the efficiency and to deep into the different fields of automation, the association is divided into thematic groups, one of them is named 'Modelling and Simulation', constituting the group.

→ [www.cea-ifac.es/wwwgrupos/simulacion](http://www.cea-ifac.es/wwwgrupos/simulacion)

→ [simulacion@cea-ifac.es](mailto:simulacion@cea-ifac.es)

- ✉ CEA-SMSG / María Jesús de la Fuente,  
System Engineering and Automatic Control department,  
University of Valladolid,  
Real de Burgos s/n., 47011 Valladolid, SPAIN

CAE - SMSG Officers	
<b>President</b>	M. À. Piera Eroles, <a href="mailto:MiquelAngel.Piera@uab.es">MiquelAngel.Piera@uab.es</a>
<b>Vice president</b>	Emilio Jimenez, <a href="mailto:emilio.jimenez@unirioja.es">emilio.jimenez@unirioja.es</a>
<b>Repr. EUROSIM</b>	Emilio Jimenez, <a href="mailto:emilio.jimenez@unirioja.es">emilio.jimenez@unirioja.es</a>
<b>Edit. Board SNE</b>	Emilio Jimenez, <a href="mailto:emilio.jimenez@unirioja.es">emilio.jimenez@unirioja.es</a>
<b>Web EUROSIM</b>	Mercedes Peres, <a href="mailto:mercedes.perez@unirioja.es">mercedes.perez@unirioja.es</a>

*Last data update December 2012*

## LSS – Latvian Simulation Society

The Latvian Simulation Society (LSS) has been founded in 1990 as the first professional simulation organisation in the field of Modelling and simulation in the post-Soviet area. Its members represent the main simulation centres in Latvia, including both academic and industrial sectors.

→ [briedis.itl.rtu.lv/imb/](http://briedis.itl.rtu.lv/imb/)

✉ [merkur@itl.rtu.lv](mailto:merkur@itl.rtu.lv)

- ✉ LSS / Yuri Merkuryev, Dept. of Modelling and Simulation Riga Technical University  
Kalku street 1, Riga, LV-1658, LATVIA

LSS Officers	
<b>President</b>	Yuri Merkuryev, <a href="mailto:merkur@itl.rtu.lv">merkur@itl.rtu.lv</a>
<b>Secretary</b>	Artis Teilans, <a href="mailto:Artis.Teilans@exigenservices.com">Artis.Teilans@exigenservices.com</a>
<b>Repr. EUROSIM</b>	Yuri Merkuryev, <a href="mailto:merkur@itl.rtu.lv">merkur@itl.rtu.lv</a>
<b>Deputy</b>	Artis Teilans, <a href="mailto:Artis.Teilans@exigenservices.com">Artis.Teilans@exigenservices.com</a>
<b>Edit. Board SNE</b>	Yuri Merkuryev, <a href="mailto:merkur@itl.rtu.lv">merkur@itl.rtu.lv</a>
<b>Web EUROSIM</b>	Oksana Sosho, <a href="mailto:oksana@itl.rtu.lv">oksana@itl.rtu.lv</a>

*Last data update December 2012*

## ROMSIM – Romanian Modelling and Simulation Society

ROMSIM has been founded in 1990 as a non-profit society, devoted to theoretical and applied aspects of modelling and simulation of systems. ROMSIM currently has about 100 members from Romania and Moldavia.

→ [www.ici.ro/romsim/](http://www.ici.ro/romsim/)

✉ [sflorin@ici.ro](mailto:sflorin@ici.ro)

- ✉ ROMSIM / Florin Stanculescu,  
National Institute for Research in Informatics, Averișcu  
Av. 8 – 10, 71316 Bucharest, Romania

ROMSIM Officers	
<b>President</b>	Florin Stanculescu, <a href="mailto:sflorin@ici.ro">sflorin@ici.ro</a>
<b>Vice president</b>	Florin Hartescu, <a href="mailto:flory@ici.ro">flory@ici.ro</a> Marius Radulescu, <a href="mailto:mradulescu@ici.ro">mradulescu@ici.ro</a>
<b>Repr. EUROSIM</b>	Florin Stanculescu, <a href="mailto:sflorin@ici.ro">sflorin@ici.ro</a>
<b>Deputy</b>	Marius Radulescu, <a href="mailto:mradulescu@ici.ro">mradulescu@ici.ro</a>
<b>Edit. Board SNE</b>	Florin Stanculescu, <a href="mailto:sflorin@ici.ro">sflorin@ici.ro</a>
<b>Web EUROSIM</b>	Zoe Radulescu, <a href="mailto:radulescu@ici.ro">radulescu@ici.ro</a>

*Last data update December 2012*



## RNSS – Russian Simulation Society

NSS - The Russian National Simulation Society (Национальное Общество Имитационного Моделирования – НОИМ) was officially registered in Russian Federation on February 11, 2011. In February 2012 NSS has been accepted as an observer member of EUROSIM.

→ [www.simulation.su](http://www.simulation.su)

✉ [yusupov@ias.spb.su](mailto:yusupov@ias.spb.su)

✉ RNSS / R. M. Yusupov,  
St. Petersburg Institute of Informatics and Automation  
RAS, 199178, St. Petersburg, 14th lin. V.O, 39

### RNSS Officers

President	R. M. Yusupov, <a href="mailto:yusupov@ias.spb.su">yusupov@ias.spb.su</a>
Chair Man. Board	A. Plotnikov, <a href="mailto:plotnikov@sstc.spb.ru">plotnikov@sstc.spb.ru</a>
Secretary	M. Dolmatov, <a href="mailto:dolmatov@simulation.su">dolmatov@simulation.su</a>
Repr. EUROSIM	R. M. Yusupov, <a href="mailto:yusupov@ias.spb.su">yusupov@ias.spb.su</a>
Deputy	B. Sokolov, <a href="mailto:sokol@ias.spb.su">sokol@ias.spb.su</a>
Edit. Board SNE	Y. Senichenkov, <a href="mailto:sneyb@dcn.infos.ru">sneyb@dcn.infos.ru</a>

*Last data update February 2012*

## LIOPHANT Simulation

Liophant Simulation is a non-profit association born in order to be a trait-d'union among simulation developers and users; Liophant is devoted to promote and diffuse the simulation techniques and methodologies; the Association promotes exchange of students, sabbatical years, organization of International Conferences, organization of courses and stages in companies to apply the simulation to real problems.

→ [www.liophant.org](http://www.liophant.org)

✉ [info@liophant.org](mailto:info@liophant.org)

✉ LIOPHANT Simulation, c/o Agostino G. Bruzzone,  
DIME, University of Genoa, Polo Savonese,  
via Molinero 1, 17100 Savona (SV), Italy



### LIOPHANT Officers

President	A.G. Bruzzone, <a href="mailto:agostino@itim.unige.it">agostino@itim.unige.it</a>
Director	E. Bocca, <a href="mailto:enrico.bocca@liophant.org">enrico.bocca@liophant.org</a>
Secretary	A. Devoti, <a href="mailto:devoti.a@iveco.com">devoti.a@iveco.com</a>
Treasurer	Marina Masseimassei@itim.unige.it
Repr. EUROSIM	A.G. Bruzzone, <a href="mailto:agostino@itim.unige.it">agostino@itim.unige.it</a>
Deputy	F. Longo, <a href="mailto:f.longo@unica.it">f.longo@unica.it</a>
Edit. Board SNE	F. Longo, <a href="mailto:f.longo@unica.it">f.longo@unica.it</a>
Web EUROSIM	F. Longo, <a href="mailto:f.longo@unica.it">f.longo@unica.it</a>

*Last data update December 2012*

## SNE – Simulation Notes Europe

Simulation Notes Europe publishes peer reviewed *Technical Notes*, *Short Notes* and *Overview Notes* on developments and trends in modelling and simulation in various areas and in application and theory. Furthermore SNE documents the ARGESIM Benchmarks on *Modelling Approaches and Simulation Implementations* with publication of definitions, solutions and discussions (*Benchmark Notes*). Special *Educational Notes* present the use of modelling and simulation in and for education and for e-learning.

SNE is the official membership journal of EUROSIM, the Federation of European Simulation Societies. A News Section in SNE provides information for EUROSIM Simulation Societies and Simulation Groups.

SNE is published in a printed version (Print ISSN 2305-9974) and in an online version (Online ISSN 2306-0271). With Online SNE the publisher ARGESIM follows the Open Access strategy, allowing download of published contributions for free. Since 2011 Online SNE contributions are identified by an DOI (Digital Object Identifier) assigned to the publisher ARGESIM (DOI prefix 10.11128). Print SNE, high-resolution Online SNE, source codes of the *Benchmarks* and other additional sources are available for subscription via membership in a EUROSIM society.

**Authors Information.** Authors are invited to submit contributions which have not been published and have not being considered for publication elsewhere to the SNE Editorial Office. SNE distinguishes different types of contributions (*Notes*):

- *Overview Note* – State-of-the-Art report in a specific area, up to 14 pages, only upon invitation
- *Technical Note* – scientific publication on specific topic in modelling and simulation, 6 – 8 (10) pages
- *Education Note* – modelling and simulation in / for education and e-learning; max. 6 pages
- *Short Note* – recent development on specific topic, max. 4 pages
- *Software Note* – specific implementation with scientific analysis, max 4 pages
- *Benchmark Note* – Solution to an ARGESIM Benchmark; basic solution 2 pages, extended and commented solution 4 pages, comparative solutions on invitation

Interested authors may find further information at SNE's website → [www.sne-journal.org](http://www.sne-journal.org) (layout templates for *Notes*, requirements for benchmark solutions, etc.).



ASIM



ASIM



ASIM



## ASIM - Buchreihen / ASIM Book Series

### **Fortschritte in der Simulationstechnik / Frontiers in Simulation Monographs - Conference Proceedings**

#### **Simulation und Optimierung in Produktion und Logistik.**

L. März, W. Krug, O. Rose, G. Weigert, G. (eds.), ISBN 978-3-642-14535-3, Springer, 2011

#### **Integrationsaspekte der Simulation: Technik, Organisation und Personal.**

Zülch, G., Stock, P. (eds.), ISBN 978-3-86644-558-1, KIT Scientific Publishing, Karlsruhe, 2010

#### **Verifikation und Validierung für die Simulation in Produktion und Logistik, Vorgehensmodelle und Techniken.**

M. Rabe, S. Spieckermann, S. Wenzel (eds.); ISBN: 978-3-540-35281-5, Springer, Berlin, 2008

#### **Advances in Simulation for Production and Logistics Applications.**

M. Rabe (ed.), ISBN 978-3-8167-7798-4, Fraunhofer IRB-Verlag, Stuttgart, 2008

#### **Modellierung, Regelung und Simulation in Automotive und Prozessautomation**

- Proceedings 5. ASIM-Workshop Wismar 2011. -

C. Deatcu, P. Dünow, T. Pawletta, S. Pawletta (eds.), ISBN 978-3-901608-36-0, ASIM/ARGESIM, Wien, 2011.

### **Reihe Fortschrittsberichte Simulation / Series Advances in Simulation**

#### Ch. Steinbrecher: **Ein Beitrag zur prädiktiven Regelung verbrennungsmotorischer Prozesse**

FBS 18, ASIM/ARGESIM Vienna, 2010; ISBN 978-3-901608-68-1, ARGESIM Report 68

#### O. Hagendorf: **Simulation-based Parameter and Structure Optimisation of Discrete Event Systems**

FBS 17, ASIM/ARGESIM Vienna, 2010; ISBN 978-3-901608-67-4, ARGESIM Report 67

#### D. Leitner: **Simulation of Arterial Blood Flow with the Lattice Boltzmann Method**

FBS16, ASIM/ARGESIM Vienna, 2009; ISBN 978-3-901608-66-7, ARGESIM Report 16

#### Th. Löscher: **Optimisation of Scheduling Problems Based on Timed Petri Nets.**

FBS 15, ASIM/ARGESIM Vienna, 2009; ISBN 978-3-901608-65-0, ARGESIM Report 15

#### J. Wöckl: **Hybrider Modellbildungszugang für biologische Abwasserreinigungsprozesse.**

FBS 14, ASIM/ARGESIM Vienna, ISBN 3-901608-64-8, 2006, ARGESIM Report 14,

#### M. Gyimesi: **Simulation Service Providing als Webservice zur Simulation Diskreter Prozesse.**

FBS 13, ASIM/ARGESIM Vienna, ISBN 3-901-608-63-X, 2006, ARGESIM Report 13

#### R. Fink: **Untersuchungen zur Parallelverarbeitung mit wissenschaftlich-technischen Berechnungsumgebungen.**

FBS 12, ASIM/ARGESIM Vienna, 2008; ISBN 978-3-901608-62-9, ARGESIM Report 12

#### H. Ecker: **Suppression of Self-excited Vibrations in Mechanical Systems by Parametric Stiffness Excitation.**

FBS 11, ASIM/ARGESIM Vienna, ISBN 3-901-608-61-3, 2006, ARGESIM Report 11

#### **Orders:**

ASIM/ARGESIM Office Germany, Hochschule Wismar, PF 1210, 23952 Wismar, Germany

ASIM/ARGESIM Geschäftsstelle Österreich, c/o DWH, Neustiftgasse 57, 1040 Vienna, Austria

Order and Download via ASIM webpage in preparation

**Info:** [www.asim-gi.org](http://www.asim-gi.org), [info@asim-gi.org](mailto:info@asim-gi.org)

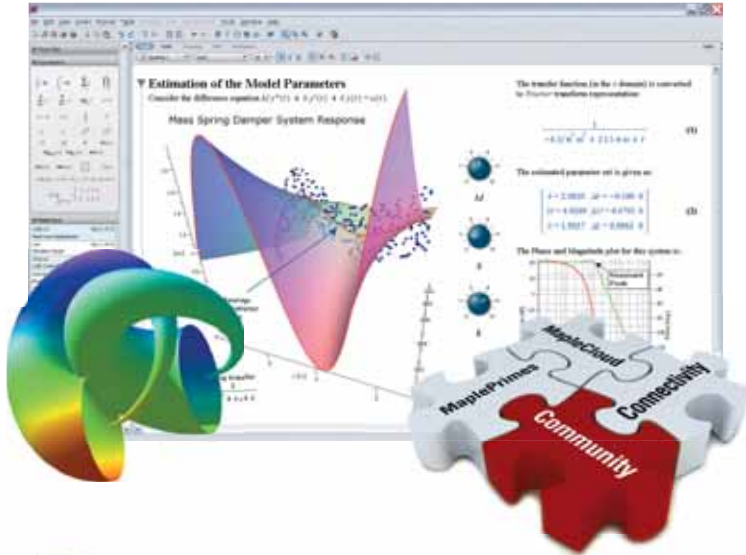


REPORTS



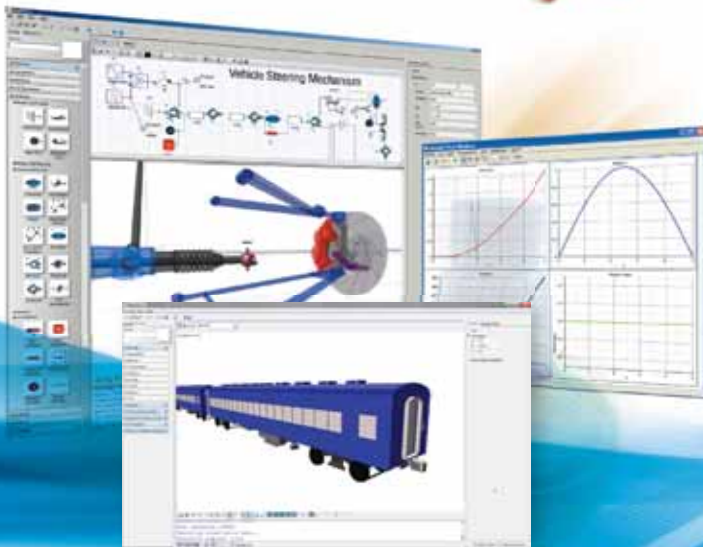
REPORTS

# New challenges in modeling and simulation require powerful tools



## Maple™ 15

The result of over 30 years of cutting-edge research and development, Maple combines the world's most powerful mathematical computation engine with an intuitive, "clickable" user interface.




## MapleSim™ 5

MapleSim is a physical modeling and simulation tool built on a foundation of symbolic computation technology.

**Get a Price Quote:**  
[www.maplesoft.com/sne](http://www.maplesoft.com/sne)

**Contact us:**

 +49 (0)241/980919-30





# EUROSIM 2013

8<sup>th</sup> EUROSIM Congress on Modelling and Simulation

The City Hall, Cardiff, Wales, United Kingdom 10-13 September 2013



EUROSIM Congresses are the most important modelling and simulation events in Europe. For EUROSIM2013, we are soliciting original submissions describing novel research and developments in the following (and related) areas of interest: Continuous, discrete (event) and hybrid modelling, simulation, identification and optimization approaches. Two basic contribution motivations are expected: M&S Methods and Technologies and M&S Applications. Contributions from both technical and non-technical areas are welcome.

## Congress Topics

The EUROSIM 2013 Congress will include invited talks, parallel, special and the poster sessions. The Congress topics of interest include, but are not limited to:

Intelligent Systems and Applications  
Hybrid and Soft Computing  
Communication Systems and Networks  
Case Studies, Emergent Technologies  
Workflow Modelling and Simulation  
Web-based Simulation  
Security Modelling and Simulation  
Computer Games and Simulation  
Neural Networks, Fuzzy Systems &  
Evolutionary Computation  
Autonomous Mental Development  
Bioinformatics and Bioengineering  
Circuits, Sensors and Devices

e-Science and e-Systems  
Image, Speech & Signal Processing  
Human Factors and Social Issues  
Industry, Business, Management  
Virtual Reality, Visualization and  
Computer Games  
Internet Modelling, Semantic Web  
and Ontologies  
Computational Finance & Economics  
Systems Intelligence and  
Intelligence Systems  
Adaptive Dynamic Programming and  
Reinforcement Learning

Methodologies, Tools and  
Operations Research  
Discrete Event /RT Systems  
Mobile/Ad hoc wireless  
networks, mobicast, sensor  
placement, target tracking  
Control of Intelligent Systems  
and Control Intelligence  
Robotics, Cybernetics, Control  
Engineering, & Manufacturing  
Energy, Power, Transport,  
Logistics, Harbour, Shipping  
and Marine Simulation  
Semantic & Data Mining

## Congress Venue / Social Events

The Congress will be held in the historic and magnificent City Hall in the heart of Cardiff, the capital city of Wales. The Gala Dinner will be held in the main hall of the National Museum of Wales. Social activities include visits to Cardiff Castle and Caerphilly Castle.

**Congress Team:** K. Al-Begain, A. Orsoni, R. Zobel, R. Cant, D. Al-Dabass; [kbegain@glam.ac.uk](mailto:kbegain@glam.ac.uk)

**Info:** [www.eurosim2013.info](http://www.eurosim2013.info)