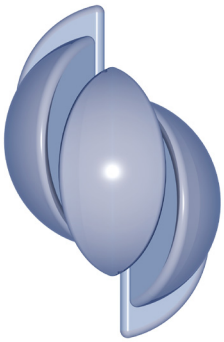


Proceedings of
SIMS 2004

the

45th International Conference of
Scandinavian Simulation Society

Copenhagen, Denmark
September 23–24, 2004

EDITORS
Brian Elmegaard, Jon Sporring, Kenny Erleben, Kim Sørensen

ORGANIZED BY
Technical University of Denmark,
Copenhagen University, and
Aalborg University

These Proceedings are available from:
Department of Mechanical Engineering
Technical University of Denmark
Nils Koppels Allé
Building 402-403
DK-2800 Kgs. Lyngby
Denmark
Tel.: (+45) 45 25 41 22
Fax: (+45) 45 93 52 15

Printed by IKON Document Services, Denmark

SIMS 2004 Logo Design by 3Dfacto, Denmark

ISBN 87-7475-316 9

Organizing Committee

B. Elmegaard	(DTU)
Kim Sørensen	(AUC)
Kenny Erleben	(KU)
Jon Sporring	(KU)

Scientific Committee

Bjarne Andresen	University of Copenhagen, Denmark
Niels Christensen	Technical University of Denmark, Denmark
Michael Damsgaard	Anybody Technology, Denmark
Jeppe Grue	Elsam, Denmark
Benny Gøbel	Technical University of Denmark, Denmark
Jesper Hattel	Technical University of Denmark, Denmark
Niels Houbak	Technical University of Denmark, Denmark
Arne Jakobsen	Instituttet for Produktudvikling, Denmark
Jakob Munch Jensen	Babcock & Wilcox Vølund, Denmark
Kaj Juslin	VTT Automation, Finland
Esko Juuso	University of Oulu, Finland
Per Lagoni	Energy Solutions, Intl., Denmark
Bernt Lie	Telemark University College, Norway
Bo Friis Nielsen	Technical University of Denmark, Denmark
Mads Pagh Nielsen	Aalborg University, Denmark
Bjørn Qvale	Technical University of Denmark, Denmark
Per Grove Thomsen	Technical University of Denmark, Denmark
Jon Agust Thorsteinsson	Marorka, Island

Arranged by

Technical University of Denmark
http://www.dtu.dk/

University of Copenhagen
http://www.ku.dk/

Aalborg University
http://www.auc.dk/

Cooperating Organizations

Scandinavian Simulation Society
http://www.scansims.org/

Dansk Simuleringsforening
http://www.dksim.dk/

Sponsors

3DFacto
http://www.3dfacto.dk

Comsol
http://www.comsol.com

Aalborg Industries
http://www.aalborg-industries.com/

IPU
http://www.ipu.dk

Otto Mønsteds fond
http://www.ottomoensted.dk/

Sim-Serv
http://www.sim-serv.com

Preface

The members of the Organizing Committee of SIMS 2004 are pleased to present the Proceedings of SIMS 2004. The SIMS 2004 conference is the 45'th annual conference of the Scandinavian Simulation Society, SIMS.
In Denmark the conference is held for the third time in a row at the Technical University of Denmark hosted by the Department of Mechanical Engineering. SIMS'95 and SIMS 2000 were also hosted here. For the first time the Danish section of SIMS, Dansk Simuleringsforening, has been involved in the organization of the conference. Dansk Simuleringsforening was formed in 2001 among others by the Department of Mechanical Engineering of the Technical University of Denmark and by Department of Energy Engineering of Aalborg University.
This year it was decided to include Computer Animation as a topic for the conference, and the Department of Computer Science at the University of Copenhagen was invited to take part in the organization. We find that the extension of the conference scope has been a success, and we conclude that the extension has strengthened the cross-disciplined nature of the conference.
The proceedings clearly show the many and different areas, where modelling and simulation are invaluable tools for the scientists and engineers. The paper handling was performed using the CyberChair system, and 85 abstracts were submitted in the areas of: Refrigeration Mechanical/Electronic Systems, Energy Systems, Manufacturing Processes, Numerical Methods, Simulation of Communication Systems, Modelling and Simulation Tools, Process Optimization and Diagnostics, Animation (Rigid- and Soft-body Simulation), Process Industry, and Gass Distribution. From these were 52 papers included in the proceedings. Geographically the authors are mainly from Scandinavia, but we are happy to find that a substantial amount of authors from all of the world have contributed. It is our hope that the participants will benefit from the cross-discipline relations that will occur at the conference.
The organizers wish to express gratitude to the large work performed by the reviewers listed elsewhere in the proceedings by supplying the authors with valuable comments and suggestions for improvements of their papers. In addition, we would like to thank the sponsors also listed elsewhere in the proceedings, without whose contributions it would not have been possible to host a conference at such a high scientific level.

We hope you enjoy the conference and the proceedings,

Brian Elmegaard, Jon Sporring, Kim Sørensen, and Kenny Erleben

Papers

Process Industry and Manufacturing Processes 1

A Sim–Serv Test Case in the UK for Simulation of Manufacturing Processes
Aldini, Lorenzo; Maccioni, Raffaele 3

Simulation of Stress–Strain Behavior for One–dimensional Aluminum Samples Subjected to High Temperature
Bellini, Anna; Thorborg, Jesper; Hattel, Jesper 7

Simulation and Control of Solidification of a Liquid Metal Column
Furenes, Beathe; Lie, Bernt 17

Modelling of an Industrial Urea Synthesis Reactor Using Sequential Modular Approach
Hamidipour, Mohsen; Mostoufi, Navid; Sotudeh-Gharebagh, Rahmat 27

Modelling the Influence of the Gas to Melt Ratio on the Fraction Solid of the Surface in Spray Formed Billets
Hattel, Jesper; Pryds, N.H. 35

Prediction of Coated Paper Properties by Artificial Neural Networks
Hladnik, Ales; Stark, Helmut; Stravs-Grilc, Barbara 45

A Big Leap Forward – the New European Cost Action E 36 “Modelling and Simulation in the Pulp & Paper Industry”
Kappen, Johannes; Dahlquist, Erik; Negro, Carlos; Manninen, Jussi; Ritala, Risto 53

Experiences and Challenges in Dynamic Simulation of Paper Production Processes
Lappalainen, Jari; Kangas, Petteri; Ylén, Jean-Peter; Savolainen, Jouni; Juslin, Kaj; Pursimo, Juha 59

Integrated Modelling of the Welding, Machining and Heat Treatment Processes
Thorborg, Jesper; Hattel, Jesper; Hald, John 69

Simulation of Capacitor Discharge Stud Welding
Weiss, Dietmar; Lauenroth, Thomas 79

Process Optimization and Diagnostics 87

Conditions for Intelligent Process Diagnostics and Optimization in the Future
Dahlquist, Erik; Dhak, Janice 89

Data Reconciliation for Real–time Optimization for an Industrial Coke–Oven–Gas Purification Process
Faber, Richard; Li, Bo; Li, Pu; Wozny, Günter 95

Model Based Control and Optimization in Pulp Industry
Jansson, Johan; Dahlquist, Erik 103

Methods for Handling Uncertainties in Dynamic Simulation and Optimization: Case TMP Plant
Pulkkinen, Petteri; Ritala, Risto 111

Energy Systems 117

Control Volume Based Modelling of Compressible Flow in Reciprocating Machines
Andersen, Stig Kildegård; Carlsen, Henrik; Thomsen, Per Grove 119

Nodal Modelling of Biomass Thermochemical Decomposition
Andrei Vijeu, Razvan; Bellettre, Jerome; Tazerout, Mohand 129

CFD Biomass Two–stage Downdraft Gasifier Modelling: Oxidation–Reduction Zone
Gerun, Luc; Vîjeu, Rãzvan; Bellettre, Jerome; Tazerout, Mohand 137

Dynamic Simulation of a Solar Collector Field with Intelligent Distributed Parameter Models
Juuso, Esko; 145

Computer–based Design Tool for Geothermal Piping Systems
Karlsdottir, Anna; Jonsdottir, Fjola; Jonsson, Magnus Thor 155

Proposal of a Simple Polymer Electrolyte Membrane (pem) Fuel Cell Stages Assessment Method
Lo Mastro, Fabio; Blando, Gandolfo 163

Modeling of Solar Radiation on Part Shaded Walls
Naghashzadegan, Mohammad; Nikian, Mohammad 173

Optimization of the Heat Exchanger Network of a Reforming System
Nielsen, Mads Pagh; Korsgaard, Anders; Kær, Søren Knudsen 181

Calculation of Heat Flow from Buried Pipes Using a Time–dependent Finite Element Model
Persson, Tommy; Wollerstrand, Janusz 189

Dynamic Modeling of Domestic Hot Water System Using Simulink
Persson, Tommy; Wollerstrand, Janusz 199

Evaluation of Operation Characteristics of the Counter–flow Cooling Tower Â– Design and Off–design Performance
Shin, Jee-Young; Son, Young-Seok; You, Byoung-Mu 207

Analysis on the Performance of Shell–and–tube Heat Exchangers with Conventional and Spiral Baffle Plates
Son, Young-Seok; Kim, Sung-Kwon; Shin, Jee-Young 215

Planning of Fuel and Energy Supply in Latvia by Using a Mesap Programming Model
Vostrikovs, Sergejs; Turlajs, Daniels; Kundzina, Antra 223

Numerical Methods 231

Numerical Simulation of Combustion in Direct Injection Diesel Engines. Application to Additional Heating Systems
Garnier, Cedric; Bellettre, Jerome; Tazerout, Mohand; Guyonvarch, Gwénaël 233

Numerical Simulation of Combined Screw Compressor–Expander Machines for Use in High Pressure Refrigeration Systems
Kovacevic, Ahmed; Stosic, Nikola; Smith, Ian. K 241

New Methods for Numerical Investigation of Dynamic Processes
Slonevsky, Roman; Stolyarchuk, Roksolana 249

A Numerical Investigation of the Effect of Inlet Position on the Transient Turbulent Flow in a Fuel Oil Storage Tank
Vardar, Nurten; 255

Mechanical/Electronic Systems 263

Analysis of Noise in Ventilating and Air–conditioning Systems
Naghashzadegan, Mohammad; Nikian, Mohammad 265

Simulated Stress and Stretch of SWCNT
Proykova, Ana; Iliev, Hristo 273
Magnetorheological Dampers in Earthquake Protection: a Simulation Study
Stammers, Charles; Guglielmino, Emanuele; D'Ayala, Dina; Sireteanu, Tudor; Stancioiu, Danut . 281

Rigid- and Soft-body Simulation 289

Simulation of Liquid–, Deformable– and Rigid–bodies
Boldt, Niels; 291
Analysis of Musculoskeletal Systems in the Anybody Modeling System
Damsgaard, Michael; Rasmussen, John; Christensen, Søren Tørholm; Surma, Egidijus; de Zee, Mark 299
Contact Graphs in Multibody Dynamics Simulation
Erleben, Kenny; Dohlmann, Henrik 307
Self–intersection with Cullide
Meyer, Jonas; Boldt, Niels 315
Ballet Balance Strategies
Pedersen, Camilla; Erleben, Kenny; Sporring, Jon 323
Real–time Simulation of Smoke Using Graphics Hardware
Rørbech, Marinus; 331

Modelling and Simulation Tools 339

Modelling of Fluid Proprieties in Hydraulic Positive Displacement Machines
Casoli, Paolo; Vacca, Andrea; Franzoni, Germano; Berta, Gian Luigi 341
Generalized Net Model of Storehouse Processes: a Principal Scheme
Christova, Nikolinka G.; Atanassov, Krassimir T. 351
Istanbul–Ankara High Speed Train Travel Analysis by Railroad Vehicle Simulation
Guney, Baha; Taymaz, Imdat; Cakir, Kemal; Gündogar, Emin 357
Modelling of a One Pass Smoke Tube Boiler
Karstensen, Claus M.S.; Sørensen, Kim 365
Fluid Flow and Particle Deposition Simulations in the Human Nose
Kleven, Marit; Melaaen, Morten; Reimers, Martin; Røtnes, Jan Sigurd; Aurdal, Lars; Djupesland, Per G. 373
Towards Automatic Generation of Model Checkable Code from Modelica
Lundvall, Håkan; Bunus, Peter; Fritzson, Peter 381
Gridmodelica – a Modelling and Simulation Framework for the Grid
Nyström, Kaj; Aronsson, Peter; Fritzson, Peter 389
The Use of a Zero Dimensional Model in Order to Characterise Knock in Spark Ignition Gas Engines
Ollivier, Eric; Bellettre, Jerome; Tazerout, Mohand 397
An Integrated Framework for Model–driven Product Design and Development Using Modelica
Pop, Adrian; Fritzson, Peter; Johansson, Olof 405
A Topology Approach to Modelling
Preisig, Heinz A; 413
Field Type and Field Constructor in Modelica
Saldamli, Levon; Fritzson, Peter 421
Modelling, Simulating and Optimizing Boilers
Sørensen, Kim; Condra, Thomas; Niels, Houbak 429

Author Index **437**

Key Word Index **439**

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

Process Industry and Manufacturing Processes

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

A SIM-SERV TEST CASE IN THE UK FOR SIMULATION OF MANUFACTURING PROCESSES

Lorenzo Aldini* and Raffaele Maccioni
Actsolutions S.r.l.,
21040 Jerago con Orago, Varese, Italy

Abstract

For this costumer placed in Belfast, UK, simulation has been used to investigate the manufacturing process, to evaluate bottlenecks and to prepare the background for simulation based, in-line scheduling. The actual problem at the shop floor deals with synchronizing work flows and optimizing production in conjunction with customer orders . Over 1700 different kinds of orders can be manufactured at the shop floor and the first task in building the model, was to separate the wide amount of different types of products into main groups. Using the simulation model in connection with more advanced scheduling tools, production managers expect significant improvements of the production efficiency, obtained by reducing the waiting time in queues and by increasing the usage of the resources.

Overview of the manufacturing at the shop floor

At the shop floor metal sheets are stored in automatic storage/retrieval system (FMS) that feeds all of the 6 punching machines. Generally products are completed on one machine however there is flexibility to allow component transfer between machines for breakdowns. After punching, the steel is returned into the FMS, then the punched parts are brought to any of 14 folding machines. All machines need an operator full time. Furthermore each machine is manned at all time with an additional operator, or trainer, per shift to help facilitate 2 man operations for some of the larger components. Next step is welding: since the work here is done manually, the capacity depends on the number of workers allocated to this section. The absolute maximum of workers working in this section at the same time is 17. Then there is a fettling process, again capacity depending on the number of workers, the maximum being 12 per shift. Next, there are two paint lines, moving at constant speed. The paint line conveyer speed can be adjusted to suit capacity requirements. The maximum number of operators required per line is 8 for loading and unloading and 4 for the painting operation. The paint lines do stop every hour to allow the changeover of personnel, also due to the complex nature of the product the paint lines are stopped on a regular basis to allow the painters to touch up the more complex products. The painted parts are then moved to two assembly lines and to a nutserting area. The personnel on the paint line stagger lunch breacks so that the line can run continuously. The parts once nutserted are then moved into and out of dispatch. Maximum number of operators in the nutserting area is 5 per shift. The two assembly lines are designed for mixed model

*corrisponding author. Phone/Fax: +39 544 986271
Lorenzo.aldini@actsolutions.it

production: the Lister line is dedicated predominantly for one range of semi-products, although a limited number of other semi-products can be accommodated on this line. The main line is used for all of the other products including prototypes. The main line has 10 stations and varies from 1 operator per station to 5 operators per station. The Lister assembly line has 8 stations, all of which are 1-man operations. Finished products are transferred onto shipping pallets and booked into stock.

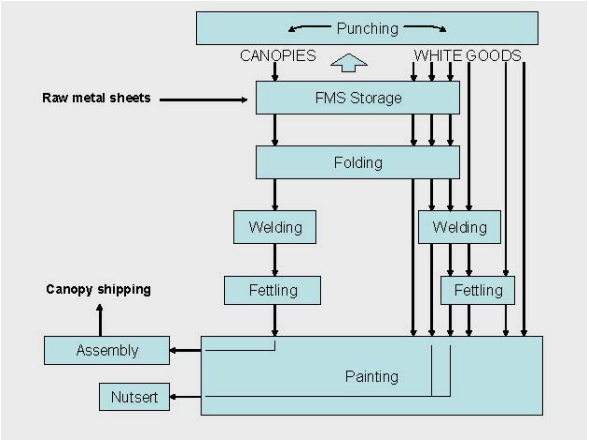


Figure 1: shop floor, work flow of the main products by routing

The problem to be solved and the role of Sim-Serv

The general management requested the plant to improve its performance with the following goals:

- Reduction in WIP by 20%
- Reduction in finished goods inventory by 25%
- Increase Plant efficiency by >5%

The plant management intended to study in a systematic way all possible losses and identify all space for improvement. They planned to identify bottlenecks and search for techniques to eliminate them; study the effects of breakdowns and absenteeism; calculate the maximum possible output that can be achieved with the existing resources and study the effect of additional resources (machines, workers); study what improvements could be achieved by better production planning and scheduling. Sim-Serv's dissemination activities had made them aware of the power of simulation models. They invited Sim-Serv's experts to visit them and discuss a possible application of this technology and how it could help them achieve the above stated goals. In the discussion, it became clear very quickly, that a simulation model would enable them to find answers to all the questions listed above, and that the financial risk of using such a model would be rather low. In co-operation with Sim-Serv experts, a specification of the simulation model and of the objectives of its use was developed, This was sent by Sim-Serv to a number of experienced simulation service providers across Europe. Within a week's time, three interesting offers were submitted by suppliers from different EU member states.They differed in price as well as with regard to the later use of the model for planning and scheduling support.
Sim-Serv assisted the costumer in the evaluation of these offers and in negotiating details with the suppliers. Finally, a decision was made, and a contract signed with the preferred supplier Actsolutions from Italy.

The approach

The functional specification developed by Sim-Serv already contained many data relevant for model development. Based on this specification, Actsolutions developed a list of more detailed data that they would need to build the model. Local technical staff in Belfast contributed by collecting all the relevant data from the factory database over a period of one month, focusing on processing times for automatic and manual procedures at workshops and on the distribution of different types of products manufactured at the shop floor, in order to feed the simulation with realistic samples. Some criteria to classify the different types of products were suggested on the basis of the data collected. The first classification distinguishes between two main

products, the so called Canopies and White Goods. Canopies are big boxes manufactured and assembled as containers for diesel engines. White Goods are smaller products that don't require assembly at the shop floor. They are mainly used as parts for control panels. There are around 1700 white goods and 60 Canopies, each Canopy made up of 40 or more parts. Each part has a unique routing through the facility. This means that all products will follow certain routing but not all products will go to each operation i.e. some products are just punched, painted and then dispatched, whereas some products will be punched, folded, welded, fettled, painted, assembled and then dispatched. White Goods have been divided into eight groups, depending on different routing (fig.1). Further classification by number of folds, number of welding points, or by painting colour have been studied to refine the model. On the side of Canopies, a good sample has been given by taking a list of the top thirty Canopies manufactured in 2003. A full day visit at the plant at mid December 2003 allowed the discussion of the data with the local staff. At a second meeting two months later, the simulation model developed by using *Arena* Rockwell Software[1] was validated with significant results. In the time between the two meetings, the work proceeded by exchanging information though the internet.

Analysis of results

While programming the simulation model, relevant production data were collected into a framework that helped to go deep in the process features. The use of the simulation model brought to better understanding of the limits of the workshops and made it possible see bottlenecks while increasing loading. Some experiments were carried out to emphasize the effects of overloads. Having tested the model in regular situation, with an average of around 72 white goods batch orders/day and 50 canopies/day, two more scenarios were tested under the following conditions: 30% increase of the number of Canopies. 30% increase of the number of White Goods. An additional situation was checked, with regular orders' flow but reduced number of operators at the workshops of welding and fettling. In each situation, *Arena* model simulated thirty days running with ten repetitions. The analysis focused on the flow times, on the usage of machines and on the queues. The most interesting results about the increase of the flow times are summarized in table 1. The increase of flow times due to bottlenecks are highlighted in red. Note that White Goods batch orders find their bottleneck at Folding, while Canopies at Welding. An apparently strange effect in the reduction of flow times for White Goods, while increasing the number of canopies, can be explained as consequence of congestion in the Canopy lines, that leaves more resources available for the White Goods. Finally, the reduction of the number of operators at welding and at fettling has a dramatic effect in the increase of process times at welding. The use of this simulation tool has been helping the fulfillment of the stated goals, as plant efficiency slightly increases by using this software, making realistic the desired goal of 5% improvement. Since it hasn't been used to support in-line scheduling yet, we don't have data to confirm the quantitative benefits required for WIP reduction and for finished goods inventory reduction.

Future developments

Real time monitoring of the flows and of the utilization of machines and of operators, are key issues for the improvement of process control. The overall efficiency depends on the reliability of the machines and on the effectiveness of the reactions to breakdowns. While the manufacturing process is running, inline scheduling can be achieved by evaluating the work load in each workshop and re-scheduling can be done when failures or unexpected events occur. It is relevant to know if any change in scheduling determines better exploitation of all the resources. A detailed

simulation model can be useful in checking alternative scenarios, planning in-line scheduling and get the best exploitation of all the resources. Both simulation and scheduling software will be connected to the SAP System. The simulation model running real-time, taking data from the workshops and from the SAP system, will check the best schedule, according to the Key Performance Indicators (KPI) accepted and shared by the management (i.e.: minimization of set-up times, of flow times, etc.). The work load, as weekly number of WG batch orders and of Canopies, will be set by production managers as well. According to these parameters the simulation model will run real-time to find out the best scheduling. Long term *steady* simulation will run background to evaluate bottlenecks and suggest improvements, about the number and the distribution of the operators at workshops, the power of machines, etc. All data required for inline scheduling will be collected from the operators and sent back to the shop floor through the PC network, supplied with field Instrumentation and connected to the SAP system.

		Folding	Welding	Fettling	Painting
Regular	W.G. Canopies	23h 28h	17h 47h	11h 30h	9h 15h
+ 30 % Canopies	W.G. Canopies	18h 36h	11h *66h*	7h 29h	7h 13h
+ 30 % W. G.	W.G. Canopies	*53h* 29h	43h 29h	14h 23h	12h 18h
- 8 workers at welding	W.G. Canopies	24h 28h	*90h* *100h*	32h 60h	5.5h 11h

Table 1: Example of flow times at workshops with regular condition, and effects of more Canopies (2nd row), more White Goods (3rd row), less workers (4th row).

Conclusion

A preliminary simulation model has been developed for this Shop floor as Sim-Serv test case. Analysis of data samples taken from the SAP system was executed to point out the processing times and allow the definition of basic categories for different kinds of orders, with focus on the sequences of workshops involved in manufacturing. The data analysis can be refined by taking a wider data sample and by including some more categories to distinguish other kinds of orders: at present, the simulation has been modelled with a degree of accuracy suitable to check the flow times as monthly averages, by changing the number of available operators or by increasing the amount of orders. This model can be used as support for monthly planning, or just for training purposes to investigate the process dynamics. The further step, after refining the model adequately, aims to connect the simulator with in-line control tools, to manage scheduling synchronize work flows and optimize production in conjunction with customer orders.

References

[1] W.D. Kelton, R.P. Sadowski, D.T. Sturrock: *Simulation with Arena*, 3rd Ed., Mc Graw Hill, 2004.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

SIMULATION OF STRESS-STRAIN BEHAVIOR FOR ONE-DIMENSIONAL ALUMINUM SAMPLES SUBJECTED TO HIGH TEMPERATURE

Anna Bellini *, Jesper Thorborg and Jesper Hattel
Technical University of Denmark
Department of Manufacturing Eng. and Management
2800 Lyngby
Denmark

Abstract

In order to satisfy the growing need in high quality aluminum cast parts of the automobile industries, in the last decades the foundries have been showing an increasing interest in the implementation of numerical simulations as part of their process design. As a consequence, it is possible to find in literature several programs capable of simulating the entire casting process, i.e. filling, solidification, as well as developed thermomechanical stresses. However, it is common practice in the foundry industry that the results obtained by the simulation of the cast process are "forgotten" during the analysis of the next phases, such as heat treatment and life prediction of the cast parts. Because of the lack of numerical program tools capable of predicting the stress-strain behavior of aluminum parts subjected to high temperature, it is indeed normally assumed that at the end of the thermal treatment the residual stresses are negligible. Nevertheless, in order to account for eventually "forgotten" thermal stresses, the automobile parts are usually over-designed.
It is the objective of this work, that is part of the IDEAL (**I**ntegrated **De**velopment Routes for Optimized Cast **Al**uminium Components) project, financed by the EU in frame work 6 and born in collaboration with the automobile and foundry industries, to fill the mentioned gap. Through a systematic analysis of experimental tests, this study aims to develop a powerful predicting tool capable of capturing stress relaxation effects through an adequate description of the creep behavior of the aluminum alloys at high temperature.

Keywords: Viscoplasticity, creep, unified constitutive model, aluminum, high temperature.

Nomenclature

a	constant of proportionality
A	constant of proportionality
B	material parameter
E	elastic modulus [Pa]
K	drag stress [Pa]
K^*	drag stress at steady state [Pa]
$\dot{K}$	evolution rate of drag stress [Pa/s]
K_0	initial drag stress [Pa]
L	length of a specimen [m]
n	exponent of the power law
Q	activation energy [j/mol K]
R	universal gas constant [j/mol K]
S_{ij}	ij-component of the deviatoric stress tensor [Pa]
t	time [s]
t_n	time at the n^{th} step [s]
T	absolute temperature [K]
T_m	absolute melting temperature [K]
ε	total strain in 1-D
ε_n	total strain at the n^{th} step in 1-D
$\dot{\varepsilon}^{in}$	inelastic strain rate in 1-D [1/s]
ε_e^{in}	effective inelastic strain
$\dot{\varepsilon}_e^{in}$	effective inelastic strain rate [1/s]

*Corresponding author. Phone: +45 4525 4739, Fax: +45 4593 4570, E-mail: anna@ipl.dtu.dk

$\dot{\varepsilon}_{ij}^{in}$	ij-component of the inelastic strain rate tensor [1/s]
σ	stress level in 1-D [Pa]
σ^*	steady state stress level in 1-D [Pa]
σ^{trial}	trial stress in 1-D [Pa]
σ_e	equivalent deviatoric stress [Pa]
σ_{ij}	ij-component of the stress tensor [Pa]
Δt	time step [s]
$\Delta\varepsilon_n$	increment in the total strain at the n^{th} step in 1-D
$\Delta\sigma_n$	increment in the stress at the n^{th} step in 1-D

Introduction

The aim of this study is the development of a numerical program for the prediction of stress-strain behavior of aluminum parts subjected to thermal treatments. Once incorporated in the preliminary design stage of the entire cast process, this simulation tool will insure continuity in the prediction of the evolution of thermomechanical stress, from filling, solidification and heat treatment to the final manufactured part.

This paper presents the work done so far for the modelling of the creep behavior of aluminum parts at high temperature in the one-dimensional case. More precisely, it shows:

- the implementation, in a computational form, of a unified constitutive model with a one-state variable;
- the methodology to follow in order to determine the material parameters required for the specific model;
- the comparison of the computed results with the experimental values obtained during the simulation of tensile and creep tests.

The importance of this one-dimensional model can be seen not only in prospective of the development of a more complex three-dimensional program to implement in a general simulation system, i.e. MAGMASOFT, but also as a tool for the extraction of the material parameters from creep and tensile tests. Since these experimental tests are basically one-dimensional, a simpler one-dimensional program is of great use for parametric studies also in the future.

Unified Constitutive Models

At high temperature metallic materials and alloys in general show a decrease in strength with an increase in temperature. At the same time the ductility increases and rate effects become more pronounced. For the modelling of rate dependent phenomena, such as viscoplasticity, as well as temperature dependent behaviors, e.g. creep and stress relaxation, unified constitutive models are often applied.

A unified constitutive model was introduced in 1976 by Miller [11] for modelling the behavior of metals at high temperature. It was then used and further developed by Anand [1] and successively many others, such as Lu et al. [10], Sehitoglu et al. [12], [13] and Smith et al. [15].

The fundamental assumption of all these models is that creep and viscoplasticity are both irreversible strains developed because of dislocations motion in the material structure; hence they can be modelled using the same constitutive laws, in terms of one or more state variables. All the mentioned models are based on the definition of two state variables: the *back stress*, responsible for the kinematic hardening, and the *drag stress*, responsible for the isotropic hardening. More specifically, the back stress is usually introduced for modelling the cycling loading; while the drag stress is introduced to take into account the hardening and softening occurred because of microstructural changes in the material, due to the temperature and to the strain rate. For the purposes of the present work, since cyclic loading is not considered, the back stress is not taken into account.

Developed Model

In order to simulate the heat treatment processes of aluminum cast parts, a simple unified constitutive model with a single state variable, the drag stress, was developed according to the following procedure.

Viscous material behavior of metals at high temperature is often modelled in literature by assuming a power law for the effective inelastic strain rate:

$$\dot{\varepsilon}_e^{in} = a * \sigma_e^n \tag{1}$$

where $\dot{\varepsilon}_e^{in}$ is the equivalent inelastic strain rate, while σ_e is the equivalent deviatoric stress:

$$\dot{\varepsilon}_e^{in} = \left(\frac{2}{3}\dot{\varepsilon}_{ij}^{in}\dot{\varepsilon}_{ij}^{in}\right)^{\frac{1}{2}}, \quad \sigma_e = \left(\frac{3}{2}S_{ij}S_{ij}\right)^{\frac{1}{2}} \tag{2}$$

In order to take into account strain hardening [10], which leads to a decreasing creep rate with increasing plastic strain, equation (1) is modified to:

$$\dot{\varepsilon}_e^{in} = a * \left(\frac{\sigma_e}{K}\right)^n \tag{3}$$

In equation (3) the parameter a can be thought as a reference strain rate. The hardening exponent n is a constant, which normally falls in the range of 32 to 200 for most metals at room temperature. Consequently the drag stress K would be equal to the effective Von Mises stress for the material, if the Mises stress were measured in a tensile test with the tensile strain rate prescribed such that $\varepsilon_{ij}^{pl} = a$ [17].
For the drag stress K the following simple evolution law (see also [1]) is considered:

$$\dot{K} = B * (K^* - K)\, \dot{\varepsilon}_e^{\,in} \tag{4}$$

where B and K^* are material constants. Since $\dot{K} = 0$ when $K = K^*$, K^* can be seen as the value of K when the steady-state is reached (i.e. during secondary creep).
The constant B controls the amount of isotropic hardening produced by a given amount of strain. As such, it plays an important role in transient situations, such as:

- when a tensile test is simulated. Higher values of B cause the stress to rise rapidly as nonelastic strain is imposed;
- when a creep test is simulated. Higher values of B lead to a more important primary creep, while lower values can result in a negligible primary creep.

By integrating equation (4), the drag stress can be expressed as:

$$K = K^* - (K^* - K_0)\, e^{-B\varepsilon_e^{in}} \tag{5}$$

where K_0 is a constant, which represents the initial value, i.e. the value of K for $\varepsilon_e^{in} = 0$.
With the purpose of taking into account the temperature dependence, the common Arrhenius law is introduced, and equation (3) is modified to:

$$\dot{\varepsilon}_e^{in} = A * e^{\left(-\frac{Q}{RT}\right)} * \left(\frac{\sigma_e}{K}\right)^n \tag{6}$$

where Q represents the activation energy, R the universal gas constant and T the absolute temperature.

If equation (6) is considered more thoroughly, it can be noticed that the model described here does not have any yield surface: a non-zero plastic rate corresponds indeed to any stress different from zero [17]. Instead of one particular yield surface the viscoplastic material has a whole family of neighboring plastic potential surfaces, which mark a gradual transition from mainly elastic to mainly plastic behavior (see figure 1).

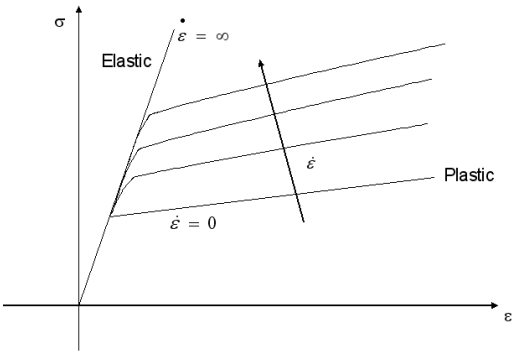


Figure 1: Typical Stress-Strain curves for a viscoplastic material

Constitutive Equations for 3D

In order to calculate all the components of the three-dimensional tensor, it is necessary to introduce a tensor relation between the strain rate tensor and the deviatoric stress tensor:

$$\dot{\varepsilon}_{ij}^{in} = \frac{3}{2} * \dot{\varepsilon}_e^{in} * \frac{S_{ij}}{\sigma_e} \tag{7}$$

where $\dot{\varepsilon}_{ij}^{in}$ is the ij component of the inelastic strain rate tensor and S_{ij} is the deviatoric stress :

$$S_{ij} = \sigma_{ij} - \frac{1}{3}\sigma_{kk}\delta_{ij} \tag{8}$$

Substituting equation (6) into equation (7) leads to:

$$\dot{\varepsilon}_{ij}^{in} = \frac{3}{2} * A * e^{\left(-\frac{Q}{RT}\right)} * \left(\frac{\sigma_e}{K}\right)^n * \frac{S_{ij}}{\sigma_e} \tag{9}$$

which is the constitutive equation, that together with the growth law (4) was implemented in a 1D numerical program.

Constitutive Equations for 1D

In the case of a one dimensional tensile test or creep test, since $\sigma_e = |\sigma_{11}|$ and $S_{11} = \frac{2}{3}\sigma_{11}$, equation (7)

becomes:

$$\dot{\varepsilon}_{11}^{in} = \frac{3}{2} * \dot{\varepsilon}_{e}^{in} * \frac{2}{3}\frac{\sigma_{11}}{|\sigma_{11}|} = \dot{\varepsilon}_{e}^{in} * sign(\sigma_{11}) \qquad (10)$$

Substituting equation (6) into equation (10) it is possible to write:

$$\dot{\varepsilon}^{in} = A * e^{\left(-\frac{Q}{RT}\right)} * \left(\frac{|\sigma|}{K}\right)^{n} * sign(\sigma) \qquad (11)$$

where the subscript 11 has been neglected for simplicity.
Since in a usual tensile test, as well as in a creep test, the applied stress is always positive, equation (12) can be simply rewritten as:

$$\dot{\varepsilon}^{in} = A * e^{\left(-\frac{Q}{RT}\right)} * \left(\frac{\sigma}{K}\right)^{n} \qquad (12)$$

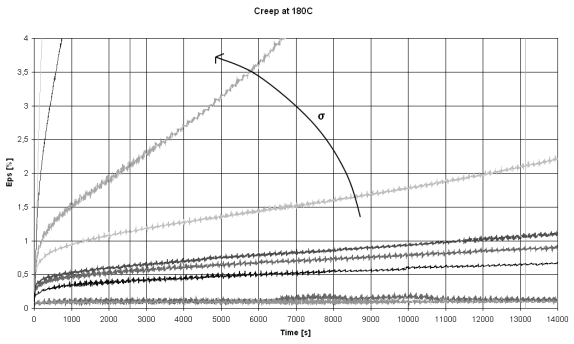


Figure 2: Experimental creep curves at 180^oC

Determination of Material Constants

In order to determine the value of the exponent n, the secondary creep phase of experimental creep curves at $T = 180^oC$ and several imposed stresses were considered (see figure 2).
For the purpose of determining the secondary creep rate, the mentioned curves were linearized as shown in figure 3. By plotting $\dot{\varepsilon}^{in}$ versus the applied stress σ^* (here the imposed stress σ is called σ^* in order to remember that those values correspond to the steady-state, i.e. secondary creep) two regions of different creep mechanisms could be recognized (see figure 4). More specifically it could be noticed that for $\sigma^* \leq 110\ MPa$ the experimental points were aligned on an exponential with smaller n (see equation (6)) with respect to the points for $\sigma^* > 110$. For this reason it was considered $K^* = 110\ MPa$ and the values of σ^* were then normalized with respect to it.

However, in order to determine the values of n and A of the equation (6), a value for Q was needed as well. As first guess, a value of $Q = 210000\ J/mol$, which corresponds to the theoretical activation energy for aluminum [5], was imposed.
Rearranging equation (6) leads to:

$$log_{10}\left(\frac{\dot{\varepsilon}^{in}}{e^{\left(-\frac{Q}{RT}\right)}}\right) = n * log_{10}\left(\frac{\sigma}{K}\right) + log_{10}(A) \qquad (13)$$

By plotting $log_{10}\left(\frac{\dot{\varepsilon}^{in}}{exp\left(-\frac{Q}{RT}\right)}\right)$ versus $log_{10}\left(\frac{\sigma^*}{K^*}\right)$ it is consequently possible to fit a linear curve, thus to determine the value of n, which represents the slope, and the value of A, i.e. the value of the intercept (see for example figure 6). Nevertheless, it should not be forgotten that the value of Q was only a guess. Consequently this value needed to be corrected, with a trial and error approach, so to fit also the experimental data at different temperatures.

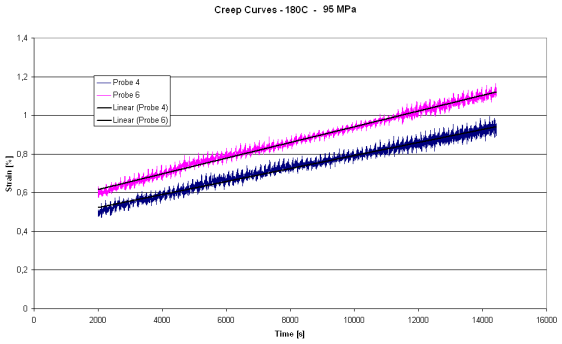


Figure 3: Determination of secondary creep rate for creep tests at 180^oC and 95 MPa. Curves for two different samples.

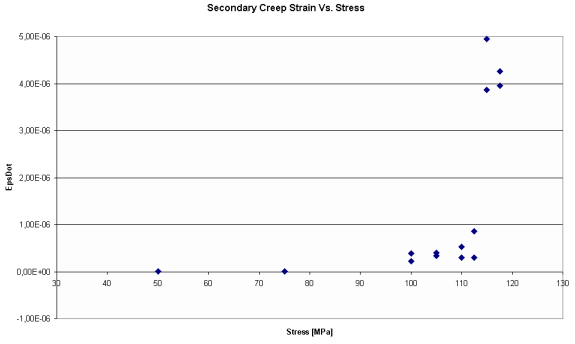


Figure 4: Plot of $\dot{\varepsilon}^{in}$ versus σ^*

Since creep data were not available, the tensile test curves at $T = 250^oC$ were considered. The program was run for $Q = 210000\ J/mol$ and the steady state part of the curve was compared with the experimental data (see figure 5). According to the result of the comparison, the value of Q was modified, and consequently also n and A were adjusted. This process was repeated till when a good agreement between the numerical and the experimental curve was found for $Q = 150\ KJ/mol$ (see figure 5).
For the new value of Q, two creep laws were determined for $\frac{\sigma}{K} \leq 1$ and $\frac{\sigma}{K} > 1$ (see figure 6):

$$\dot{\varepsilon}^{in} = \begin{cases} \bar{A} * e^{\left(-\frac{150}{RT}\right)} \left(\frac{\sigma}{K}\right)^{\bar{n}_1} & \text{for } \left(\frac{\sigma}{K}\right) \leq 1 \\ \bar{A} * e^{\left(-\frac{150}{RT}\right)} \left(\frac{\sigma}{K}\right)^{\bar{n}_2} & \text{for } \left(\frac{\sigma}{K}\right) > 1. \end{cases} \tag{14}$$

where $\bar{A}, \bar{n}_1$ and $\bar{n}_2$ are constants that can not be reported here because of confidential reasons. Once the material parameters Q, n and A were determined, a value of K_0 had to be found in order to best simulate the experimental data. For this reason the tensile experimental curves at different temperatures and different strain rates were considered. The value of the yield stress (σ_0) and the corresponding ($\dot{\varepsilon}^{in}$) were extrapolated for each curve and a value of K_0 was determined, for each curve, as:

$$K_0 = \left(\frac{A * e^{\left(-\frac{Q}{RT}\right)}}{\dot{\varepsilon}^{in}} \right)^{\frac{1}{n}} * \sigma_0 \tag{15}$$

where it was considered $\bar{n}_1$, because strain rates $\dot{\varepsilon}^{in} > 1e-7$ (see figure 6) are realistic in tensile tests. An average value of $K_0 = 77\ MPa$ was then calculated.
The last value still to be determined was the hardening parameter B, that, as said before, plays an important role during the transient situations, both in the stress-strain as well as in the creep curves. A best-fit value of B was found by simulating and then comparing both tensile tests and creep tests at different temperatures and different strain rates. It was concluded that for $\frac{\sigma}{K} \leq 1$ the hardening parameter B was a constant $B = 250$; while for $\frac{\sigma}{K} > 1$ a temperature dependent hardening parameter was more accurate. In conclusion, it was considered that

$$B = \begin{cases} 210 & \text{for } \left(\frac{\sigma}{K}\right) \leq 1 \\ 1.94 * T - 520 & \text{for } \left(\frac{\sigma}{K}\right) > 1. \end{cases} \tag{16}$$

where T is the absolute temperature. Using this set of data, the results shown in the "Results" section were obtained.

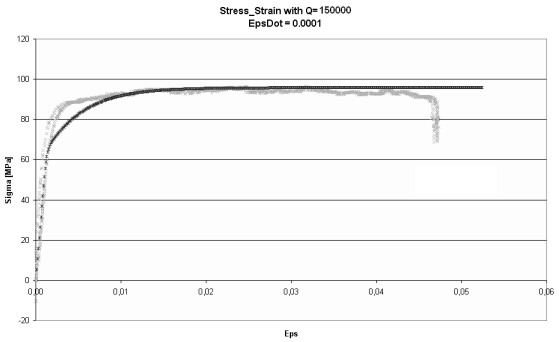


Figure 5: Determination of Q: comparison between experimental and numerical curve for a tensile test at $T = 250^oC$

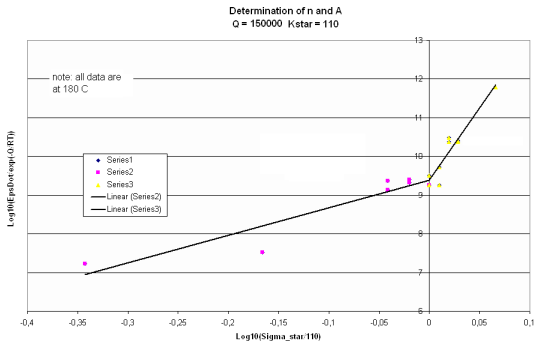


Figure 6: Determination of the parameters n and A

Remarks

In the model described in the previous sections and implemented in the mentioned numerical code, the activation energy Q and the initial drag stress K_0 were considered constant in all the range of temperatures. However, two observations need to be acknowledged:

- according to the theory of the dislocation motion [11] Q can be considered constant only for temperatures $T \geq 0.6T_m$, where T_m is the melting temperature of the alloy.

- K_0 is usually considered temperature dependent in literature. Nevertheless in this study it was seen that the stress-strain curves were quite

well reproduced using a constant value (see figure 8). In a future improvement of the program, it will also be considered to have K_0 dependent on the temperature. In this case it is obvious that more experimental curves will be needed.

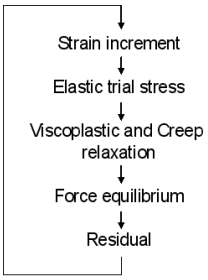


Figure 7: Numerical implementation of the viscoplastic-creep algorithm

Numerical Implementation in 1D

In order to determine all the material parameters needed for the model, a one dimensional program was developed at first. It is indeed from tensile and creep tests, i.e. typical one dimensional tests, that those parameters can be evaluated.
For the purpose of enhancing stability, the algorithm for the program was based on a *backward Euler* implementation in the frame work of a strain driven problem. In the algorithm (see also figure 7) a strain increment is calculated and an elastic trial state is then evaluated, similarly to the procedure described by Simo [14] for the *return mapping algorithm*; consequently an *inelastic relaxation* is calculated. The force equilibrium is then imposed and a residual is evaluated. If the residual is found to be different than zero, a new strain increment is calculated and all the procedure is repeated, applying a Newton-Rapson equilibrium algorithm.

Elastic Trial State

In order to implement a model capable of simulating inelastic relaxation, the following considerations were taken into account:

- $x \in [0, L]$ is a point of a body for which the inelastic relaxation has to be determined.

- At the current time t_n the state of the point x is totally defined, hence:

$$\left\{\varepsilon_n(x), \varepsilon_n^{in}(x)\right\} \tag{17}$$

and consequently

$$\sigma_n(x) = E_n\left[\varepsilon_n(x) - \varepsilon_n^{in}(x)\right] \tag{18}$$

are known.

- An *increment* $\Delta\varepsilon_n(x)$ is *given* at the time t_n, so to drive the state to the time $t_{n+1} = t_n + \Delta t$.

In order to determine the new state at time t_{n+1}, an auxiliary state is defined at first. This state, which does not correspond to an actual state, is defined by *freezing the inelastic flow*. This means that the given $\Delta\varepsilon_n(x)$ is considered totally elastic in the first iteration, hence a purely elastic step is calculated as follows:

$$\begin{aligned} \sigma_{n+1}^{trial} &:= E_{n+1}\left(\varepsilon_n + \Delta\varepsilon_n - \varepsilon_n^{in}\right) \\ \varepsilon_{n+1}^{trial} &:= \varepsilon_n^{in} \end{aligned} \tag{19}$$

Since the trial state is defined only in terms of the known state $\left\{\varepsilon_n(x), \varepsilon_n^{in}(x)\right\}$ and of the given $\Delta\varepsilon_n$, it can be calculated directly. However, since it is defined by *freezing the inelastic flow*, it does not correspond to an actual state.

Inelastic Relaxation

Once the trial elastic state is calculated, an inelastic relaxation, due to viscoplasticity and creep, must be evaluated.
Applying Hooke's law, it is possible to write:

$$\sigma_{n+1}(x) = E_{n+1}\left(\varepsilon_n^{el}(x) + \Delta\varepsilon_n - \Delta\varepsilon_n^{in}(x)\right) \tag{20}$$

Remembering the definition of elastic trial stress (see equation (19)) $\sigma_{n+1}^{trial} = E_{n+1}\left(\varepsilon_n^{el}(x) + \Delta\varepsilon_n\right)$, equation (20) can be rewritten as:

$$\sigma_{n+1}(x) = \sigma_{n+1}^{trial} - E_{n+1}\Delta\varepsilon_n^{in}(x) \tag{21}$$

For the determination of $\sigma_{n+1}(x)$ it is thus necessary to calculate the inelastic strain increment during the interval Δt.
Since $\Delta\varepsilon^{in} := \Delta\dot{\varepsilon}^{in} * \Delta t$, the inelastic strain increment can be evaluated using equation (11):

$$\Delta\varepsilon_n^{in}(x) = A * e^{\left(-\frac{Q}{RT}\right)} * \left(\frac{|\sigma|}{K}\right)^n * \Delta t * sign(\sigma) \tag{22}$$

Substituting equation (22) into (21) leads to:

$$\sigma_{n+1}(x) = \sigma_{n+1}^{trial} - E * A * e^{\left(-\frac{Q}{RT}\right)} * \left(\frac{|\sigma|}{K}\right)^{n} * \Delta t * sign(\sigma) \quad (23)$$

which can be rearranged in the following expression:

$$f(\sigma_{n+1}) = \sigma_{n+1}(x) - \sigma_{n+1}^{trial} + E * A * e^{\left(-\frac{Q}{RT}\right)} * \left(\frac{|\sigma|}{K}\right)^{n} * \Delta t * sign(\sigma) = 0 \quad (24)$$

From equation (23) it is evident that the stress level at the new time t_{n+1} can not be found in closed form. A Newton-Raphson algorithm was thus implemented locally in order to determine the zero of the function f of equation (24).
It can be noticed that the function f very much resembles the *effective stress function* defined by Bathe in [4]. Obviously, as suggested also by Bathe, other methods, such as the bisection technique, could be used for the evaluation of its zero.

Results

This section is dedicated to the comparison of the numerical curves with the experimental data obtained both for tensile and creep tests. About the experimental data it should be underlined that the curves show large scattering (see figure 9), probably because of the difficulties that rose for measurements at high temperatures. Consequently it is hard to compare experimental and numerical curves. For simplicity, in figure 8 for example only one experimental curve for each temperature was shown.

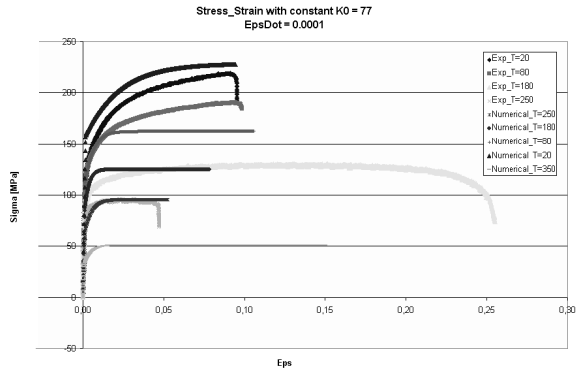


Figure 8: Tensile test curves, experimental and numerical, for different temperatures

The numerical simulations and the experimental curves obtained at several temperatures during *tensile tests* are shown in figure 8. It can be noticed that good agreement is reached for temperatures $\geq 250^{o}C$ and for the test conducted at room temperature.

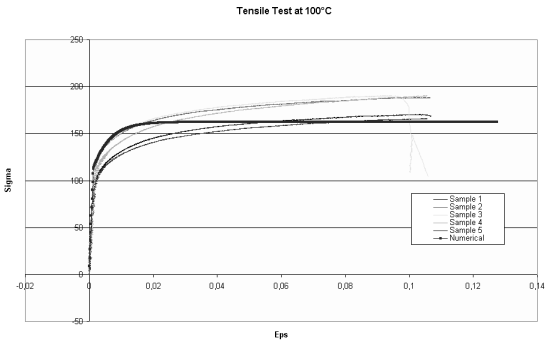


Figure 9: Experimental data for tensile test curves for $T = 80^{o}C$

For the simulation at $T = 80^{o}C$ and $T = 180^{o}C$, on the contrary, it looks like the hardening parameter B could maybe be further optimized. Nevertheless for $T = 180^{o}C$ the comparison with creep tests shows that with the applied hardening parameter the simulations well reproduce the creep curves.
For the test performed at $T = 100^{o}C$ it seems also that the stress level at steady state is underestimated. However, if also other samples are considered (see figure 9), it can be observed that a stress level of 160 *MPa* for the steady state is actually in the range of the experimental data.
The computed results and the experimental values obtained during the simulation of *creep tests* at $T = 180^{o}C$ and different stress levels are shown in figures 10 - 11. From the comparison it is possible to conclude that the developed program well predicts the creep strains during high temperature exposure of the aluminum parts for almost all the stress levels. However, it was observed that for an imposed stress of $\sigma = 75$ *MPa* the simulated curves over-estimate the strain rate during secondary creep. This result is due to the linear fitting, shown in figure 6, that was used for the determination of the exponent n. From the same figure it can be seen indeed that for

$$\sigma = 75\ MPa, \text{ hence } log_{10}\left(\frac{\sigma}{K^*}\right) = -0.16$$

the curve fitting amply over-estimates the strain rate. In order to eliminate this mismatch, a third creep

mechanism could be considered, so that:

$$\dot{\varepsilon}^{in} = \begin{cases} A_1 * e^{\left(-\frac{Q}{RT}\right)} * \left(\frac{\sigma}{K}\right)^{n_1} & \text{for } \left(\frac{\sigma}{K}\right) \leq a \\ A_2 * e^{\left(-\frac{Q}{RT}\right)} * \left(\frac{\sigma}{K}\right)^{n_2} & \text{for } a < \left(\frac{\sigma}{K}\right) \leq b \\ A_3 * e^{\left(-\frac{Q}{RT}\right)} * \left(\frac{\sigma}{K}\right)^{n_3} & \text{for } \left(\frac{\sigma}{K}\right) > b \end{cases} \tag{25}$$

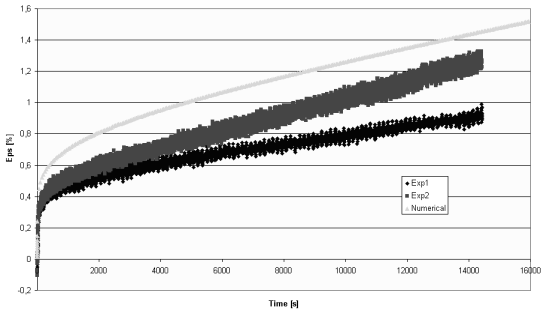


Figure 10: Comparison with a creep test conducted at $T = 180^oC$ with imposed stress $\sigma = 100\ MPa$

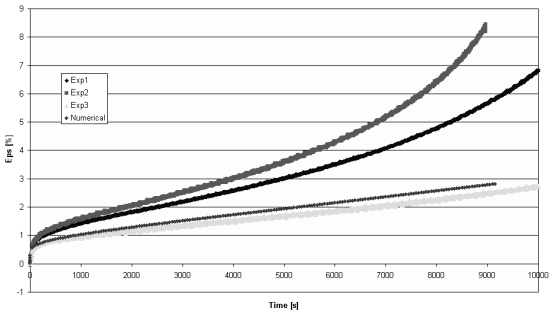


Figure 11: Comparison with a creep test conducted at $T = 180^oC$ with imposed stress $\sigma = 110\ MPa$

Few words should be spent also for the results shown in figure 11. The two experimental curves with higher strain rates show the same shape and the same range of strain rates experienced for an imposed stress of 120 *MPa*. Consequently it seems reasonable to consider that the experimental curve with lower strain rate is the most realistic. Hence, also in this case the numerical simulation well reproduces the experimental data.

Conclusions

The paper illustrates the work done for the purpose of predicting stress relaxation of aluminum at high temperature. In particular, it presents the unified constitutive model used to describe the creep behavior of aluminum. The model is based on a single state variable, the drag stress, which represents the isotropic hardening of the material. It is shown that by applying a simple growth law, see equations (4) and (16), not only the steady state, but also the transient situations can be properly modelled (see the previous section). For the simulation of all the phases of the heat-treatment of aluminum, which can last between 2 and 8 hours, not only the secondary, but also the primary creep needs indeed to be accurately predicted. With respect to other models found in the literature (i.e. [11], [12], [13]), the one presented in this paper requires fewer material parameters ($B, Q, K_0, K*, A$ and n), hence a reduced set of experimental tests.

In the applied model the initial drag stress K_0 as well as the activation energy Q were considered constant in the range of temperature under study. The reasons for these assumptions are basically based on the lack of experimental data that are needed to further describe an evolution of these two parameters with temperature. However, the results, both in terms of tensile test as well as of creep test, show that even with these simplified assumptions the model is capable of reproducing the experimental curves.

A further improvement could nevertheless be considered for the exponent n. As mentioned in the previous section, the fitting of two linear curves in the data shown in figure 6 might be too simplistic. This is the reason for the over-estimation of the secondary creep rate for the simulation of creep test at $T = 180^oC$ with a stress level of 75 *MPa*. In order to reduce the mismatch, a third linear fitting could be introduced with the purpose of modelling a third creep mechanism, for the lower stress levels.

In conclusion, the paper has described the one-dimensional program that, if used together with the presented standard procedure, allows to determine the set of material parameters need to describe the creep behavior of aluminum samples subjected at high temperatures. Once evaluated by applying the simpler one-dimensional code, the material parameters can be used in a more complex three-dimensional thermomechanical simulation analysis in order to predict the stress evolution during thermal treatments. By incorporating this "new developed tool", it would be possible in the future to guar-

antee continuity in the simulation and prediction of stresss-strain evolution in complex structure during casting, solidification, heat treatments etc., until the end of the manufacturing process.

References

[1] Anand L. *Constitutive equations for the rate-dependent deformation of metals at eleveted temperatures*, Journal of Engineering Materials and Technology 1982; 104: 12-17.

[2] Alberg H. *Material Modelling for Simulation of heat treatment*, Licentiate Thesis, Luleå University of Technology, Department of Apllied Physics and Mechanical Eng. 2003.

[3] Alberg H., Berglund D. *Comparison of plastic, viscoplastic and creep models with modelling welding and stress relief heat treatment*, Computer methods in applied mechanics and engineering 2003, 192: 5189-5208.

[4] Bathe K. J. *Finite Element Procedures*. Upper Saddle River ,NJ: Prentice Hall, 1996.

[5] Frost H. J., Ashby M. F. *Deformation- Mechanism Maps – The plasticity and creep of metals and ceramics*. Pergamon Press, 1982.

[6] Ho K., Krempl E. *Modeling of Positive, Negative and Zero Rate Sensitivity by Using the Viscoplasticity Theory Based on overstress (VBO)*, Mechanics of Time-Dependent Materials 2000; 4: 21-42.

[7] Kojic M., Bathe K. *The ´effective-stress-function´ algorithm for thermo-elasto-plasticity and creep*, International Jour. for numerical methods in engineering 1987; 24: 1509-1532.

[8] Krempl E. *Relaxation behavior and modeling*, International journal of Plasticity 2001; 17: 1419-1436.

[9] Lemaitre J., Chaboche J. *"Mechanics of solid materials*. Cambridge: Cambridge University Press, 1990.

[10] Lu Z., Weng G. *A simple unified theory for the cyclic deformation of metals at high temperature*, Acta Mechanica 1996; 118: 135-149.

[11] Miller A. *An inelastic constitutive model for monotonic, cyclic, and creep deformation: part I - Equation development and analytical procedures*. Paper No 75-Mat-14 in: Conference on Micromechanical Modelling of Flow and Fracture, 1976.

[12] Sehitoglu H., Smith T., Maier H. *Thermomechanical deformation of Al319-T7B with small secondary dendrite arm spacing*, Thermo-mechanical behavior of material: third volume, ASTM STP 1371, Americal Society for Testing and Material, 2000: pp 53-68.

[13] Sehitoglu H., Qing X., Smith T., Maier H., Allison J. *Stress-Strain response of a cast 319-T6 aluminum under thermomechanical loading*, Metallurgical and Material Transactions A 2000; 31A: 139-151.

[14] Simo J. C., Hughes T. J. R. *Computational inelasticity*. New York: Springer, 1997.

[15] Smith T., Maier H., Sehitoglu H., Fleury E., Allison J. *Modelling high-temperature stress-strain behavior of cast aluminum alloys*, Metallurgical and Material Transactions A 1999; 30A: 133-146.

[16] Tsakmalis Ch. *Formulation of viscoplasticity laws using overstresses*, Acta Mechanical 1996; 115: 179-202.

[17] Tvergaard V. *Plasticity and creep in structural materials*, Technical University of Denmark, Department of Mechanical Engineering, 2001.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

Simulation and Control of Solidification of a Liquid Metal Column

Furenes,B. & Lie, B.*
Telemark University College
P.O. Box 203, N-3901 Porsgrunn, Norway

August 15, 2004

Abstract

In this paper, two different 1D mechanistic models for the liquid-solid phase transition are presented. The first model is based on the two-domain approach, and results in 2 partial differential equations (PDEs) and one ordinary differential equation (ODE) with 2 boundary conditions, 2 interface conditions and one initial condition, namely the Stefan problem. The PDEs are discretized by use of the collocation method, and here the resulting model consists of 3 nonlinear ODEs.

In the second model, the metal column is considered as one-domain, and one PDE is valid for the whole domain. The result is one PDE with two boundary conditions. The PDE is discretized by the method of lines.

The models are implemented in `MATLAB`, and `ode23s` is used for solving the systems of equations. The models are developed in order to simulate and control the dynamic response of the solid/liquid interface velocity. The control scheme is based on a linear PI controller.

Keywords: Dynamic modeling, simulation, discretization, control, Stefan problem, phase transition

Nomenclature

Dimensional variables and parameters:

A	cross sectional area $[\mathrm{m}^2]$
$\hat{c}_p$	heat capacity $[\mathrm{J/K\,kg}]$
h	solid/liquid interface position $[\mathrm{m}]$
k	thermal conductivity $[\mathrm{W/m\,K}]$
L	heigth (length) of column $[\mathrm{m}]$
$\hat{H}$	specific enthalpy $[\mathrm{J/kg}]$
q	heat flux $[\mathrm{W/m}^2]$
Q	heat $[\mathrm{W}]$
t	time $[\mathrm{s}]$
T	temperature $[\mathrm{K}]$
v_m	solid/liquid interface velocity $[\mathrm{m/s}]$
z	space variable $[\mathrm{m}]$
α	heat diffusion coeffi cient$[\mathrm{m}^2/\mathrm{s}]$
$\Delta\hat{H}_f$	heat of fusion $[\mathrm{J/kg}]$
ρ	density $[\mathrm{kg/m}^3]$

Dimensionless variables and parameters:

c	coeffi cient in trial solution
e	controller deviation
N	number of discretization elements
K	controller parameter
R	residual for trial solution
s	position
U	temperature
u	manipulated variable
v	velocity
x	state
γ_f	heat of fusion
κ	heat diffusion coeffi cient
ζ	space variable
τ	time
F	heat

Sub-/Superscripts:

$*$	trial solution
ℓ	liquid phase
L	top of mold
s	solid phase
ζ	location in space
0	bottom of mold

*Corresponding author:Bernt.Lie@hit.no

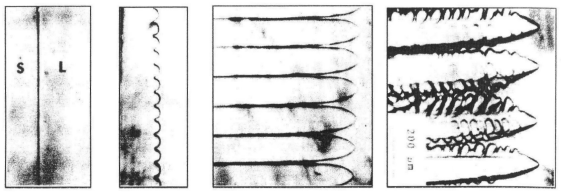


Figure 1: Example of morphology of the solid/liquid interface at different growth velocities in a transparent organic system. Taken from [2].

1 Introduction

Solidification processing involves a complex interaction between many physical mechanisms. To address modeling within solidification processing, a multidisciplinary approach must be followed. Subjects included in solidification processing may be material processing, transport phenomena, phase transition, control and optimization theory, numerical mathematics, and computer science.

It is desireable to control some of the physical mechanisms during the solidification process, i.e interfacial thermodynamics, convection, electromagnetic effects, diffusion etc. Different physcial effects influence the properties of the resulting solidified metal. Two important properties that determine the quality of the finished material are the growth velocity and the local thermal conditions at the solidification front. As an example, the concentration of impurities in the solidified metal may be dependent of the solidification velocity. At low solidification rates, the solutes (impurities) have time to diffuse away from the solid/liquid interface into the bulk liquid. At high solidification rates, the liquid phase may become undercooled, resulting in a formation of crystals in the bulk liquid. These may develop to become dendritic, resulting in segregation of impurities between the dendrite arms [1]. Thus it may be desirable to control the rate of solidification to a predefined rate to achieve a solidified metal as pure as possible. Figure 1 shows an example of the morphology of the interface as a function of growth velocity.

Solidification modeling is reported in numerous papers and books. Most of the models, however, are used for off-line simulations of complex phenomena. Many studies have been carried out in order to describe phenomena at meso- or microscale levels. Such phenomena may be morphology development, dendritic growth pattern, or mushy zone effects. It is beyond the scope of this paper to develop a model which includes the behaviour at lower levels. Modeling for control purposes involves simplifying complex processes and yet maintain a realistic behaviour of the system. The model presented in this paper is intended to be used in real-time control, and thus a fast and simple model will be pursued. According to [3], little research has been carried out on modeling for real time control purposes of directional solidification processes.

Two main approaches are distinguished in the modeling of phase transitions: the two-domain approach (also called the sharp interface approach), and the the one-domain approach (also referred to as the entalphy method). The most common method is the two-domain approach [4], in which a discontinuity of the system variables at a sharp interface is assumed. The two-domain approach results in the Stefan problem which has been studied for more than 100 years. The moving boundary between the solid and liquid phase makes the Stefan problem non-trivial from both a mathematical and a numerical viewpoint [4]. Analytical solutions are only available for a limited number of cases [5], and thus in most cases numerical solutions must be considered. Reported work on the sharp interface approach is e.g. found in [6]. For thorough information about the Stefan problem, the books [7], [8] and [9] are recommended.

In the one-domain approach, it is assumed that all the termodynamical parameters vary continuously between the two phases. One of the advantages of the one-domain approach is that the mathematically difficult problem of having a boundary at the interface (which is part of the unknown solution), is avoided. The position of the interface is determined a posteriori, from the numerical solution carried out in the fixed domain. However, one disadvantage is that an accurate location of the position may be diffi cult to obtain [9].

Control of solidification processes is reported in [10] and [11]. In these works, the control problem is adressed as an inverse problem. In several other publications, the term "inverse problems" is used to describe the design solidification problem. In this paper, the objective is to control the cooling and heating conditions at the fixed boundaries in order to achieve desired growth velocities and freezing interface heat

fluxes. Often in an inverse problem, a perfect model is assumed, and no corrections are made due to model error and disturbances. The presence of model and parameter uncertainties are taken into account in [12] where strategies to minimize a deviation error is calculated online, and action is taken by the furnace control system. Inverse Stefan problems are reported in several papers and books. [13] includes detailed information about this subject. Other works treating the inverse Stefan problem are [14], [15], [16], and [17].

This paper is organized as follows: In section 2 the solidification process is described and some assumptions are given. In section 3 the models are presented and discretized. Possible parameter values are shown in 4, and also some simulations are carried out. Control of the solidification velocity is treated in section 5, and finally some conclusions are presented in section 6.

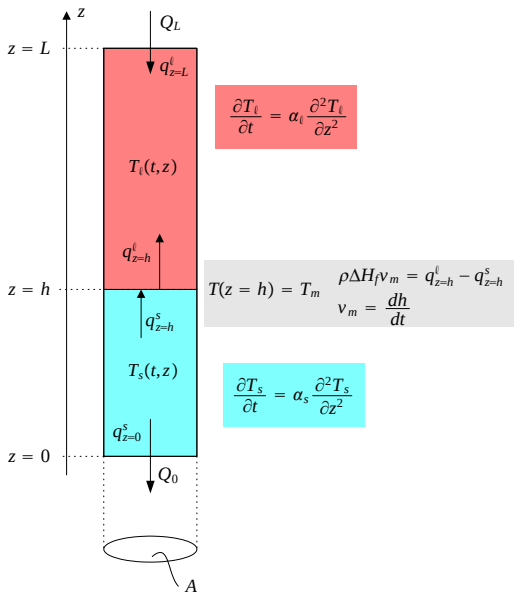


Figure 2: Sketch of solidifying metal column.

2 Process Description

We consider a metal column of height L, originally in liquid form. We want to study the solidification of the column into a solid metal rod. The system is sketched in figure 2. For simplicity, we assume that the heat flow is one-dimensional. For a casting that has a cross-sectional area much smaller than its surface area, this approximation seems to be reasonable [1].

The metal column is assumed to be fully insulated on the vertical surface. A heat source is placed at the top of the column, whereas a heat sink is placed at the bottom of the column. The constraint for maximum growth velocity is the cooling rate, and the maximum growth velocity is achieved when there is no heating at the top of the column.

3 Model

3.1 Levels of description

In solidification modeling, different levels of description exist [2], see figure 3. The different levels of modeling involve different length-scales. In a macroscopic model, the system is visible with a microscope using ordinary light and is of the order of mm. The advantage of a macroscopic model is that the gross behaviour of the system can be studied, and the fundamental physics laws can be applied to the bulk matter. In a microscopic model, however, the mechanical or chemical phenomenon is studied at an atomic or molecular level, and would not reveal the gross behaviour of the system. The order of this scale is 10^{-6} to 10^{-5} m. At the mesoscopic level there is not a sharp interface dividing the liquid and the solid, but a region between the phases. This region is called the mushy zone. The order of the mesoscopic scale is approximately one tenth of the macroscopic order. The smallest length-scale is at the nano-level at which growth kinetics and nucleation is described by the transfer of individual atoms from the liquid to the solid. The scale is of order 10^{-9} m. One of the limitations to the development of nano-scale models is the hardware limitations which does not allow computation of properties of the atomic scale in applied casting engineering.

The model developed in this paper is described at the macroscopic level. At smaller levels, the computation time will be too high for the model to be used in real-time control [18].

3.2 A two-domain approach

The model is based on the heat diffusion equation, and there is one boundary condition and one interfa-

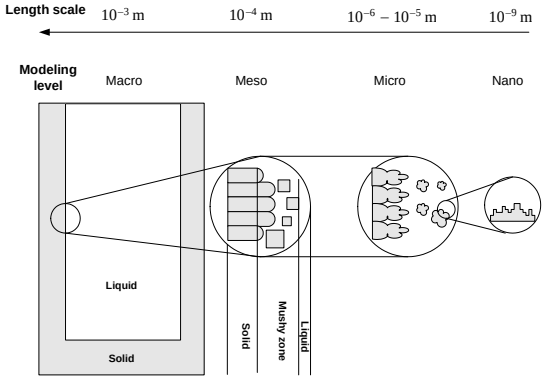


Figure 3: Different levels of description in solidification modeling. After [2].

cial condition for both solid and liquid phases. The independent variables are time t and position z, and the resulting model is a system of two PDEs and one ODE.

The spatial domain is split into two subdomains by the interface, one for each phase. We then have:

$$\text{Solid phase} \quad : \quad \frac{\partial T_s}{\partial t} = \alpha_s \frac{\partial^2 T_s}{\partial z^2} \qquad \text{for } z < h(t)$$

$$\text{Liquid phase} \quad : \quad \frac{\partial T_\ell}{\partial t} = \alpha_\ell \frac{\partial^2 T_\ell}{\partial z^2} \qquad \text{for } z \geq h(t) \quad (1)$$

where

$$\alpha_s = \frac{k_s}{\rho_s \hat{c}_p^s} \text{ and } \alpha_\ell = \frac{k_\ell}{\rho_\ell \hat{c}_p^\ell}.$$

There are two boundary conditions (one for the fixed boundary and one for the moving boundary) for each phase. For solid phase, the boundary conditions are

$$\begin{aligned} q_{z=0}^s A &= -A k_s \left.\frac{\partial T_s}{\partial z}\right|_{z=0} = -Q_o \qquad (2) \\ T_s(t,h) &= T_m. \end{aligned}$$

Corresponding, for liquid phase we have

$$\begin{aligned} q_{z=L}^\ell A &= -A k_\ell \left.\frac{\partial T_\ell}{\partial z}\right|_{z=L} = -Q_L \qquad (3) \\ T_\ell(t,h) &= T_m. \end{aligned}$$

At the interphase between the solid and liquid phases, the following energy balance is valid [19]:

$$\rho \Delta \hat{H}_f \cdot \frac{dh}{dt} = q_{z=h}^\ell - q_{z=h}^s, \qquad (4)$$

Table 1: Non-dimensional variables and parameters.

Non-dimensional variables	Non-dimensional parameters
$\zeta = \frac{z}{L}$	$\gamma_f = \frac{\Delta \hat{H}_f}{\hat{c}_p^s T_m}$
$\tau = \frac{k_s t}{\hat{c}_p^s \rho L^2}$	$\kappa_s = \frac{k_s/k_s}{\hat{c}_p^s/\hat{c}_p^s} = 1$
$U_{s,\ell} = \frac{T_{s,\ell}}{T_m}$	$\kappa_\ell = \frac{k_\ell/k_s}{\hat{c}_p^\ell/\hat{c}_p^s} = \frac{k_\ell}{k_s}$
$s = \frac{h}{L}$	
$\mathcal{F}_{0,L} = \frac{Q_{0,L} L}{k_s A T_m}$	

where:

$$\begin{aligned} q_{z=h}^s &= -k_s \left.\frac{\partial T_s}{\partial z}\right|_{z=h} \\ q_{z=h}^\ell &= -k_\ell \left.\frac{\partial T_\ell}{\partial z}\right|_{z=h}. \end{aligned}$$

It is common to present the Stefan problem in dimension-less variables. The proposed transformation in [9] is adopted in this work, see table 1.

The heat diffusion equations then become:

$$\begin{aligned} \frac{\partial U_s}{\partial \tau} &= \kappa_s \frac{\partial^2 U_s}{\partial \zeta^2} = \frac{\partial^2 U_s}{\partial \zeta^2} \\ \frac{\partial U_\ell}{\partial \tau} &= \kappa_\ell \frac{\partial^2 U_\ell}{\partial \zeta^2} \end{aligned}$$

assuming $\hat{c}_p^\ell = \hat{c}_p^s$. At the interface we get

$$\kappa_s \left.\frac{\partial U_s}{\partial \zeta}\right|_{\zeta=s(\tau)} - \kappa_\ell \left.\frac{\partial U_\ell}{\partial \zeta}\right|_{\zeta=s(\tau)} = \gamma_f \frac{ds}{d\tau}$$

where $\gamma_f = \frac{\Delta H_f}{\hat{c}_p^s T_m}$ is dimensionless latent heat.

The boundary conditions at the interface are

$$U_s|_{\zeta=s(\tau)} = U_\ell|_{\zeta=s(\tau)} = 1.$$

3.2.1 Discretization

We choose to discretize the model using the collocation method. To do so, we postulate the following trial solutions of U_s and U_ℓ:

$$\begin{aligned} U_s^*(\tau,\zeta) &= c_0^s(\tau) + c_1^s(\tau)\zeta + c_2^s(\tau)\zeta^2, \; \zeta < s(\tau) \\ U_\ell^*(\tau,\zeta) &= c_0^\ell(\tau) + c_1^\ell(\tau)\zeta + c_2^\ell(\tau)\zeta^2, \; \zeta \geq s(\tau). \end{aligned}$$

Rephrased trial solution It is desireable to have time varying coeffi cients with a physical interpretation instead of c_j^i. We thus introduce the requirements as follows:

Solid phase: We let

$$\begin{aligned} 1 &= U_s^*(\tau, s) = c_0^s + c_1^s s + c_2^s s^2 \\ F_0 &= \frac{\partial U_s^*}{\partial \zeta} = (c_1^s + 2c_2^s \zeta)_{\zeta=0} = c_1^s. \end{aligned}$$

In addition, we choose to denote the temperature of the solid at $\zeta = \zeta_s$ by U_{ζ_s}:

$$U_{\zeta_s} = U_s^*(\tau, \zeta_s) = c_0^s + c_1^s \zeta_s + c_2^s \zeta_s^2$$

where ζ_s can be a constant or a function of time. Using these relationships, we can rewrite the requirements for the solid phase as follows:

$$\begin{pmatrix} 1 & s & s^2 \\ 0 & 1 & 0 \\ 1 & \zeta_s & \zeta_s^2 \end{pmatrix} \begin{pmatrix} c_0^s \\ c_1^s \\ c_2^s \end{pmatrix} = \begin{pmatrix} 1 \\ F_0 \\ U_{\zeta_s} \end{pmatrix}. \quad (5)$$

Calculating the coeffi cients by solving (5), we can write the trial solution as

$$\begin{aligned} U_s^* &= \frac{s\zeta_s(s-\zeta_s)F_0 + \zeta_s^2 - s^2 U_{\zeta_s}}{\zeta_s^2 - s^2} \\ &\quad + F_0\zeta + \frac{(s-\zeta_s)F_0 + U_{\zeta_s} - 1}{\zeta_s^2 - s^2}\zeta^2 \end{aligned}$$

Liquid phase: Similarly, we find:

$$\begin{aligned} U_\ell^* &= \frac{\zeta_\ell s(\zeta_\ell - s)\tilde{F}_L + s(s-2)U_{\zeta_\ell}}{(s+\zeta_\ell-2)(s-\zeta_\ell)} \\ &\quad \frac{+\zeta_\ell(2-\zeta_\ell)}{(s+\zeta_\ell-2)(s-\zeta_\ell)} \\ &\quad + \frac{\left(s^2-\zeta_\ell^2\right)\tilde{F}_L + 2\left(U_{\zeta_\ell}-1\right)}{(s+\zeta_\ell-2)(s-\zeta_\ell)}\zeta \\ &\quad + \frac{(\zeta_\ell - s)\tilde{F}_L + \left(1-U_{\zeta_\ell}\right)}{(s+\zeta_\ell-2)(s-\zeta_\ell)}\zeta^2. \end{aligned}$$

where

$$\tilde{F}_L = F_L \frac{k_\ell}{k_s}.$$

Residuals In the collocation method, we force the residuals to be zero at a number of so-called *collocation* points.

In our rephrasing of the trial solution, we have made sure that the the trial solution complies with the boundary conditions. Thus, the only unknowns are U_{ζ_s} and U_{ζ_ℓ}. This means that we can only choose one collocation point for each phase.

The chosen collocation points for the liquid and solid phases are

$$\begin{aligned} \zeta_s &= \beta s \\ \zeta_\ell &= s + \lambda(1-s) \end{aligned}$$

where β and λ are positive quantities such that

$$\beta < 1$$

and

$$\lambda \ll 1.$$

The collocation points move as the solid/liquid interface evolves, and thus U_{ζ_s} and U_{ζ_ℓ} describe the temperature in the solid and the liquid phase, respectively, close to the interface.

We now get the equation for the dimension-less temperature for the solid phase in the vicinity of the interface:

$$\frac{dU_{\zeta_s}}{d\tau} = 2\frac{-s(1-\beta)F_0 + \left(1-U_{\zeta_s}\right)}{s^2\left(1-\beta^2\right)}. \quad (6)$$

The dimension-less temperature for the liquid phase in the vicinity of the interface is given by:

$$\frac{dU_{\zeta_\ell}}{d\tau} = 2\kappa_\ell \frac{-(s-1)\lambda\tilde{F}_L + \left(1-U_{\zeta_\ell}\right)}{(s-1)^2\lambda(2-\lambda)}. \quad (7)$$

For the dimension-less position we get:

$$\begin{aligned} \frac{ds}{d\tau} &= \frac{1}{\gamma_f}\left(\frac{-s(1-\beta)^2 F_0 + 2\left(1-U_{\zeta_s}\right)}{s\left(1-\beta^2\right)}\right. \\ &\quad \left. -\kappa_\ell \frac{-(s-1)\lambda^2\tilde{F}_L + 2\left(1-U_{\zeta_\ell}\right)}{(s-1)\lambda(2-\lambda)}\right) \quad (8) \end{aligned}$$

3.2.2 Total model

The total model is given by (6), (7) and (8), where the non-dimensional variables and parameters are given in table 1.

3.2.3 Numerical issues

The model is singular when either $s = 0$ or $s = 1$; with pure liquid or pure solid. It is assumed that initially we will most likely have liquid phase, only. Thus, until solidification starts, U_{ζ_s} and U_{ζ_ℓ} are not

well defined and $s = 0$. In this work a one-phase (liquid) model discretized by the method of lines is used initially, and when the temperature at position $U_\zeta = 0$ in the liquid reaches 1 (the melting temperature), the solidification model is started. Since $s = 0$ at start, the position is set to $\max(s, s_m)$ where $s_m = 10^{-5}$. Similarly, the solidification model can switch to a one-phase (solid) model when $s = \mu = 0.97$ in order to avoid the singularity at $s = 1$ in the model.

3.3 A one-domain approach

The enthalpy method in [20] is used as a basis for the one-domain model. The idea is to reformulate the heat diffusion equations for each phase in terms of the enthalpy, the sum of specific and latent heats. Then the equations (1) and (4) reduce to the single equation

$$\rho \frac{\partial \hat{H}}{\partial t} = k \frac{\partial^2 T}{\partial z^2} \tag{9}$$

where $\hat{H}$ is specific enthalpy, and it is assumed that k and ρ are independent of the temperature. The temperature and enthalpy are related by the function

$$\hat{H}(T) = \begin{cases} \hat{c}_p^s T, & T \le T_m \\ \hat{c}_p^s T_m + \Delta\hat{H}_f + \hat{c}_p^\ell (T - T_m) & T > T_m \end{cases} \tag{10}$$

and inversely, we find the temperature as

$$T\left(\hat{H}\right) = \begin{cases} \frac{\hat{H}}{\hat{c}_p^s}, & \hat{H} \le \hat{H}_l \\ T_m, & \hat{H}_l < \hat{H} < \hat{H}_u \\ T_m + \frac{\left(\hat{H} - \Delta\hat{H}_f - \hat{c}_p^s T_m\right)}{\hat{c}_p^\ell}, & \hat{H} > \hat{H}_u \end{cases}$$

where

$$\begin{aligned} \hat{H}_l &= \hat{c}_p^s T_m \\ \hat{H}_u &= \hat{c}_p^s T_m + \Delta\hat{H}_f. \end{aligned}$$

The method of lines is used to discretize (9) in the spatial domain, leading to a model that consists of ODEs describing the enthalpy as a function of time at different positions. The fixed boundary conditions specified in (2) and (3) are applied at the boundaries.

In several publications, the front position is calculated a posteriori (off-line). In a position control problem, however, an online estimat of the front must be available. A way to estimate the front is to assume a constant rate at which the liquid soldifies during the latent heat release. A function relating the fraction of solid and the enthalpy is introduced, see figure 4:

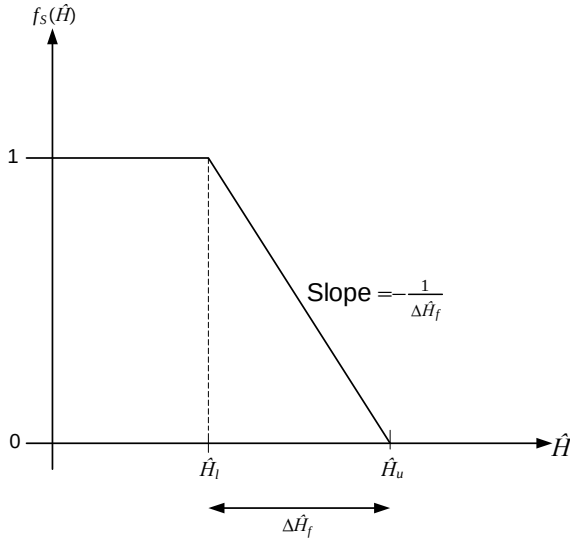


Figure 4: The fraction of solid as a function of specific enthalpy, $\hat{H}$, assuming a constant solidification rate during the latent heat release.

$$f_S(\hat{H}) = \begin{cases} 1, & \hat{H} \le \hat{H}_l \\ -\frac{1}{\Delta\hat{H}_f}\left(\hat{H} - \hat{H}_l\right) + 1, & \hat{H}_l < \hat{H} < \hat{H}_u \\ 0, & \hat{H} > \hat{H}_u \end{cases}$$

An online estimate of the position can be calculated by

$$h(t) = \sum_{i=1}^{N} f_S\left(\hat{H}_i(t)\right) \cdot \Delta z \tag{11}$$

where N is the number of discretization elements.

4 Model Analysis

4.1 Parameter values

Possible parameter values are indicated in table 2.

4.2 Simulations

The developed models were simulated with constant values for F_L and F_0. Figure 5 shows the response of the interface position for the two models. The figure shows that for the two-domain model the front moves with a constant velocity when F_L and F_0 are

Table 2: Possible parameters for the solidification model.

Parameter	Value
ρ_s, ρ_ℓ	$2.34 \times 10^3\,\mathrm{kg/m^3}$
$\hat{c}_p^s, \hat{c}_p^\ell$	$10^3\,\mathrm{J/(K\,kg)}$
k_s	$31.4\,\mathrm{W/(m\,K)}$
k_ℓ	$66.9\,\mathrm{W/(m\,K)}$
T_m	$1685\,\mathrm{K}$
$\Delta\hat{H}_f$	$1.79 \times 10^6\,\mathrm{J/kg}$
$T(t=0,z)$	$1710\,\mathrm{K}$

Table 3: Computation times for the models.

Two-domain	One-domain
3.7 s	$N = 20$: 3.4 s
	$N = 30$: 8.9 s
	$N = 40$: 19.3 s
	$N = 300$: 1800 s

constant. For the one-domain model, the velocity is higher at the start, and then decreases gradually. The model order of the one-domain model is much higher than for the two-domain model, and hence it is probably the most accurate model. The two-domain model, however, is the fastest model. The computation times for the simulation of 5000 s on a 1.59 GHz Pentium M computer with 1Gbyte RAM are shown in table 3.

5 Control

In order to control the concentration of impurities in the solidified metal, the solidification velocity must be controlled. The models developed above give the position at the interface. If we want to control the velocity, the position must track a ramp. We then need an integrator in the controller to achieve zero error at steady state.

5.1 Control of the two-domain model

We now write the model as

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{u}) \\ y &= h(\mathbf{x}, \mathbf{u}) \end{aligned} \tag{12}$$

where

$$\mathbf{x} = \begin{pmatrix} s & U_{\zeta_s} & U_{\zeta_\ell} \end{pmatrix}^T = \begin{pmatrix} x_1 & x_2 & x_3 \end{pmatrix}^T$$

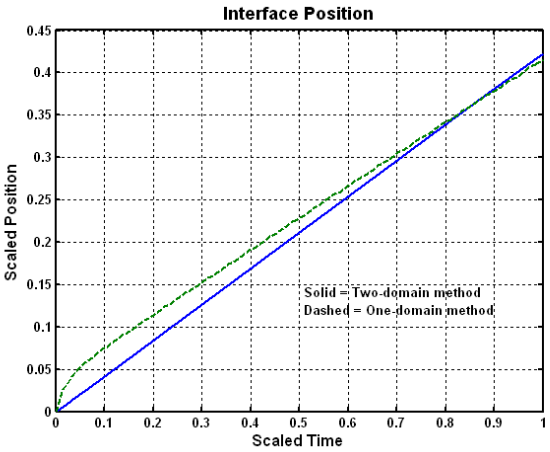


Figure 5: Simulation of the interface position for the two different models at constant heating and cooling. The number of discretization elements, $N = 40$, for the one-domain method.

and

$$\mathbf{u} = \begin{pmatrix} F_0 & F_L \end{pmatrix}^T = \begin{pmatrix} u_1 & u_2 \end{pmatrix}^T.$$

f_1, f_2 and f_3 are given by substituting x and u into the expressions in (6), (7) and (8). We also have

$$y = x_1.$$

Augmenting the state equation with the integrator $\dot{\sigma} = e = s_{ref} - x_1$, we obtain

$$\begin{aligned} \dot{\mathbf{x}} &= f(\mathbf{x}, \mathbf{u}) \\ \dot{\sigma} &= \dot{x}_4 = s_{ref}(\tau) - x_1 = f_4(\mathbf{x}, \mathbf{u}) \end{aligned}$$

where $s_{ref}(\tau) = v_{ref} \cdot \tau$ is the reference position.

The linear PI controller is given by

$$u_2 = K_P e + K_I \sigma$$

The closed loop model is simulated with the linear controller with $K_P = 20$ and $K_I = 2$. The results are shown in figure 6 (first subplot) and 7 (solid line). According to the figures, the system tracks the reference well in this particular case.

5.2 Control of the one-domain model

The controller for the one-domain model is implemented in a similar way as for the two-domain model. The controller parameters are $K_P = 15 \cdot 10^6\,\mathrm{W/m}$

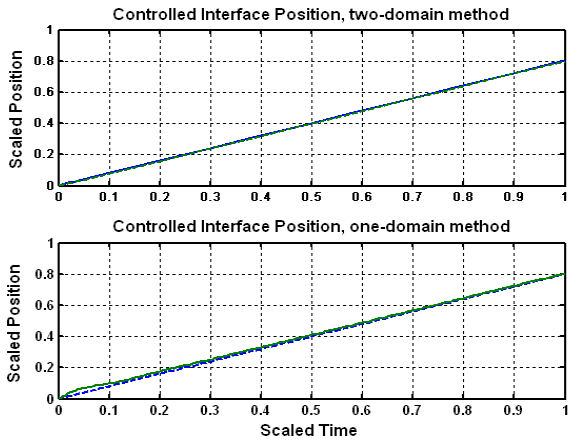


Figure 6: Controlled interface position for the two models when the dimensionless velocity reference is 0.8. The dashed lines are reference velocities, and the solid lines are estimated interface position.

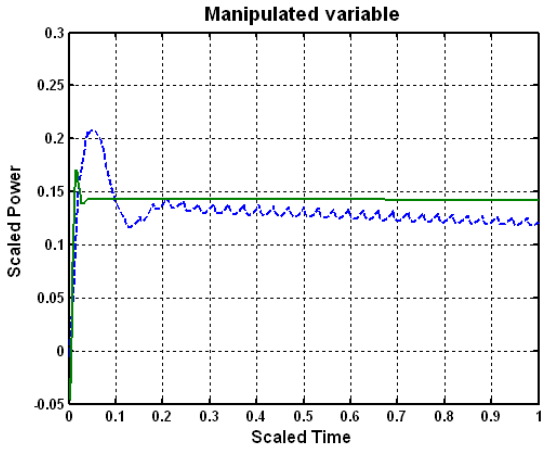


Figure 7: Manipulated variable (heating at the top of the column) for the simulated cases above. Solid line is for the two-domain model, and dashed line is for the one-domain model.

and $K_I = 4000\,\mathrm{W/m\,s}$. (The one-domain model is not dimensionless).

The manipulated variable in figure 7 (dashed line) shows a jagged behaviour. This is a discretization problem. When the number of discretization elements is increased, the manipulated variable is smoothed.

6 Conclusions

The intention of this paper is to develop two fast and simple mechanistic models for the position of the solidification interface. The models are simplified to make it suitable for control purposes, and are used to develop a linear PI-controller in order to control the solidification velocity.

The two-domain approach results in a model consisting of two PDEs and one ODE. The PDEs are discretized by the collocation method. The one-domain approach results in a model with N (number of discretizing elements) ODEs when discretized by the method of lines.

The discretized systems are implemented in MATLAB. A linear PI-controller is implemented on the systems in order to control the dynamics of the solidification front. For the cases simulated above, the position tracks the reference well.

Numerical difficulties are present in the two-domain model due to the singular behaviour of the discretized system. By introducing dimensionless variables, the numerical performance of the model is improved. The one-domain model shows no numerical difficulties. It will probably be easier to implement for 2D and 3D models since the collocation method gets very complicated for higher dimensions. A disadvantage of the one-domain model is if the enthalpy-temperature relation is more complicated, e.g. the specific heat capacities are temperature dependent. Then it may be a problem to solve the inverse of the function in (10) to get the temperature-enthalpy relation.

In conclusion, the main contributions of this paper are:

- The development of two fast and simple models for an on-line tracking of the solidification interface.
- A linear control strategy to control the solidification velocity.

Further research will include validating with plant data. Also, the models should be expanded to include 2 or 3 spatial dimensions. Incorporation of the dynamics at the top and bottom of the column will probably enhance the model further. Other modeling methods (e.g. the level set method) may also

improve the computation speed and accuracy of the front position.

References

[1] Kou, S., *Transport Phenomena and Materials Processing*, John Wiley and Sons, Inc., 1996.

[2] Stefanescu, D. M., *Science and Engineering of Casting Solidification*, Kluwer Academic/Plenum Publishers, 233 Spring Street, New York, N.Y 10013, 2002.

[3] Zabaras, N., "Inverse Techniques for the Design and Control of Solidification and Forming Processes," *Proceedings of the Integration of Material, Process and Product Design*, 1999, pp. 249–260.

[4] Benes, M., "Mathematical and Computational Aspects of Solidification of Pure Substances," *Acta. Math. Univ. Comenianae*, Vol. LXX, 2001, pp. 123–151.

[5] Flemings, M. C., *Solidification Processing*, McGraw-Hill Inc., 1974.

[6] Catalina, A. V. and Stefanescu, D. M., "A Numerical Model for the Tracking of the Solid-Liquid Interface," *Proceedings of Modeling of Casting and Solidification Processes IV*, 1999.

[7] Gupta, S. C., *The Classical Stefan Problem: Basic Concepts, Modelling and Analysis*, Elsevier Science B.V, Sara Burgerhartstraat 25, P.O Box 211, 1000 AE Amsterdam, The Netherlands, 2003.

[8] Visintin, A., *Models of Phase Transitions*, Birkhauser Boston, 675 Massachusetts Avenue, Cambridge, MA 02139, USA, 1996.

[9] Crank, J., *Free and Moving Boundary Problems*, Oxford University Press, Walton Street, Oxford OX2 6DP, 1984.

[10] Zabaras, N., "Inverse Finite Element Techniques for the Analysis of Solidification Processes," *International Journal for Numerical Methods in Engineering*, Vol. 29, 1990, pp. 1569–1587.

[11] Zabaras, N., Mukherjee, S., and Richmond, O., "An Analysis of Inverse Heat Transfer Problems With Phase Changes Using an Integral Method," *Journal of Heat Transfer*, Vol. 110, 1988, pp. 554–561.

[12] Franke, D., Steinbach, I., Krumbe, W., Liebermann, J., and Koch, W., "Concept for an Online Material Quality Control by Numerical Simulation of a Silicon Ingot Solidification Process," *Proc. 25th IEEE Photovoltaic Specialist Conference*, 1996, pp. 545–548.

[13] Goldman, N. L., *Inverse Stefan Problems*, Kluwer Academic Publishers, P.O Box, 3300 AA Dordrecht, The Netherlands, 1997.

[14] Hoffmann, K. H. and Sprekels, J., "Real-Time Control of the Free Boundary in a Two-Phase Stefan Problem," *Numerical Functional Analysis and Optimization*, Vol. 5, 1982, pp. 47–76.

[15] Niezgódka, M., "Control of Parabolic Systems with Free Boundaries - Application of Inverse Formulations," *Control and Cybernetics*, Vol. 8, 1979, pp. 213–225.

[16] Saguez, C., "Simulation and Optimal Control of Free-Boundary Problems," *Workshop on Numerical Treatment on Free Boundary Value Problems*, Vol. 58, Birkhäuser, 1982, pp. 270–286.

[17] Jochum, P., "The Numerical Solution of the Inverse Stefan Problem," *Numerische Mathematik*, Vol. 34, 1980, pp. 411–429.

[18] Ohshima, M., "Control and Design Problems in Material Processing - How Can Process Systems Engineers Contribute to Materials Processing?" *Journal of Process Control*, Vol. 13, 2003, pp. 599–605.

[19] Davis, S., *Theory of Solidification*, Cambridge Monographs on Mechanics, 2001.

[20] Voller, V. and Cross, M., "Accurate Solutions of Moving Boundary Problems Using the Enthalpy Method," *International Journal of Heat and Mass Transfer*, Vol. 24, 1980, pp. 545–556.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

Modelling of an Industrial Urea Synthesis Reactor Using Sequential Modular Approach

Mohsen Hamidipour, Navid Mostoufi and Rahmat Sotudeh-Gharebagh*
Process Design and Simulation Research Centre
Department of Chemical Engineering, University of Tehran
P.O. Box 11365/4563, Tehran, Iran

Abstract

Urea is an important product of petrochemical plants, which is mainly used as fertilizer. In this study, a model is developed based on the sequential modular simulation of chemical processes. In the proposed model the urea reactor is divided into several continuously stirred tank reactors (CSTR). In order to model the performance of these reactors the hydrodynamic and reaction submodels should be integrated together. The heterogeneous reaction of formation of ammonium carbamate was considered in this model. This reaction was considered to occur in the liquid phase in the previous works presented in literature. Also formation of biuret in the reactor is considered which has not been considered in previous works. The validity of the proposed model was demonstrated using the industrial data. The agreement between the results of the model and the industrial data was found to be satisfactory.

Keywords: Reactor modelling, Urea reactor, Heterogeneous reaction, Sequential modular approach

Nomenclature

a	moles of water added per mole of ammonium carbamate
C_{Ao}	initial concentration of A [kmol.m^{-3}]
C_{U0}	initial concentration of urea [mol.lit^{-1}]
H_I	enthalpy of inlet stream [kJ.hr^{-1}]
H_O	enthalpy of outlet stream [kJ.hr^{-1}]
k	rate constant of urea production [hr^{-1}]
k'	rate constant of biuret production [lit.mol^{-1}.hr^{-1}]
K_p	equilibrium constant of ammonium carbamate production [atm^3]
m	initial moles of carbon dioxide [mol]
n	initial moles of ammonia [mol]
n_t	total number of moles [mol]
N/C	nitrogen to carbon mass ratio
P	total pressure [atm]
P_c	critical pressure [kPa]
P_{CO2}	partial pressure of carbon dioxide [atm]
P_{NH3}	partial pressure of ammonia [atm]
Q_{ac}	heat produced by ammonium carbamate formation [kJ.hr^{-1}]
Q_b	heat consumed by biuret formation [kJ.hr^{-1}]
Q_u	heat consumed by urea formation [kJ.hr^{-1}]
$-r_A$	reaction rate [mol.m^{-3}.hr^{-1}]
t	time [hr]
T	temperature [K]
T_c	critical temperature [°C]
T_f	final temperature of reactor [ºC]
T_{NBP}	normal boiling point temperature [ºC]
V	reactor volume [m^3]
v_o	volume flow of feed [m^3.hr^{-1}]
x	moles of ammonium carbamate [mol]
X_A	partial conversion of A
X_1	partial conversion of ammonium carbamate to urea
X_2	partial conversion of urea to biuret
X_U	concentration of urea in the reactor liquid outlet stream [Wt%]
y	moles of urea per volume that reacts in t interval [mol.lit^{-1}]
ρ	density [kg.m^{-3}]

*Corresponding author. Phone: +98 21 6967781, Fax: +98 21 6957784, E-mail: sotudeh@ut.ac.ir

Introduction

Urea (NH_2CONH_2) is produced at industrial scale by the reaction between ammonia and carbon dioxide at high pressure (13-30 MPa) and high temperature (170-200 ºC). The overall reaction is as follows:

$$2NH_3 + CO_2 \Leftrightarrow NH_2CONH_2 + H_2O \quad (1)$$

The process of urea formation consists of two sequential steps. In the first step, ammonium carbamate is formed by the following reaction in the liquid phase:

$$2NH_3(l) + CO_2(l) \Leftrightarrow NH_2CO_2NH_4(l) \quad (2)$$

This reaction is very exothermic and fast in both directions so that it could be considered at equilibrium at the conditions found in industrial reactors where the residence time is rather high. In the next step, ammonium carbamate is dehydrated to form urea:

$$NH_2CO_2NH_4(l) \Leftrightarrow NH_2CONH_2(l) + H_2O(l) \quad (3)$$

This reaction is endothermic and slow as compared to the previous reaction. Therefore it needs a long time to reach the equilibrium.

There are different types of processes to produce urea in the commercial units. These processes are typically called once through, partial recycle and total recycle [1, 2]. In the total recycle process, which is employed widely, all the ammonia leaving the synthesis section is recycled to the reactor and the overall conversion of ammonia to urea is reaches 99%. Stamicarbon and Snomprogetti processes [3] are the most common examples of such process [4].

Since urea has became almost the most widely used fertilizer and its production is important in the petrochemical industry, there has been many attempts to model and simulate the reactor of urea production as the heart of the process [1-4]. Although all these researchers have considered the presence of two phases (i.e., gas and liquid) in their model, none of them have considered the existence of the following heterogeneous reaction between carbon dioxide and ammonia in the ammonium carbamate formation step:

$$2NH_3(g) + CO_2(g) \Leftrightarrow NH_2CO_2NH_4(l) \quad (4)$$

It is worth noting that in the present work, this heterogeneous reaction occurs in the reactor instead of the homogeneous reaction (2). Moreover, in the studies reported in the literature, biuret ($NH_2CONHCONH_2$) formation, which is the main undesired by-product in the urea production process, is neglected. As biuret is toxic to plants, its content in fertilizers has to be kept as low as possible. The reaction of biuret formation is:

$$2NH_2CONH_2 \Leftrightarrow NH_2CONHCONH_2 + NH3 \quad (5)$$

This is a slow, endothermic reaction. Biuret formation takes place when there is a high urea concentration, low ammonia concentration and high temperature. It has been tried in this work to model and validate the industrial urea reactor considering the above mentioned reactions.

Model Development

In this section, the hypotheses and necessary equations for developing the steady state model of the urea reactor are described in detail. A complete list of the components involved in the process of urea synthesis as well as their physical properties are shown in Table 1.

Hypotheses

Modelling the urea reactor in this work is based on the following assumptions:

- Only CO_2, H_2O, NH_3 and inert gases (O_2, N_2) exist in the gas phase.
- Formation of urea takes place only in the liquid phase.
- Dissolution of inert gases in liquid phase is neglected.
- Biuret is produced in the reactor.

Thermodynamics

Wilson and ideal gas equations were used as equations of state for liquid and gas phases, respectively. Although, the equations based on the activity were examined to predict the behavior of liquid phase the nearest results to the real data were found with using the Wilson equation. The binary interaction coefficients of Wilson equation were modified to fit the actual data. Although several equations of state were examined for the

gas phase the best results in comparison to plant data were obtained with using this couple of equations.

Component	T_{NBP} [°C]	ρ [kg.m^{-3}]	T_c [°C]	P_c [kPa]
Biuret	598.3	1068.9	770.52	997.31
Ammonium Carbamate	600	1100	785.27	1103.92
Urea	191.85	1230	431.85	9050
CO_2	-78.55	825.34	30.95	7370
H_2O	100	997.99	374.15	22120
NH_3	-33.45	616.07	132.4	11276.9
O_2	-182.95	1137.68	-118.38	5080.02
N_2	-195.8	806.37	-146.96	3394.37

Table 1. Properties of process components

Hydrodynamic Submodel

The feed to the urea reactor consists of two liquid (ammonium carbamate rich) and vapor (CO_2 and NH_3) streams, entering from bottom. This makes both phases to move upward. The movement of bubbles through the liquid phase causes mixing in the liquid phase. Moreover, there are several perforated plates at different levels inside the reactor in order to prevent back mixing and further mixing between the two phases. Consequently, the reactor can be conveniently considered as a sequence of CSTRs.

Reaction Submodel

Three main reactions considered in the process are formation of ammonium carbamate (4), formation of urea (3) and formation of biuret (5). The residence time in the urea reactor is high enough in order the reaction of ammonium carbamate formation to be practically considered at equilibrium [5]. There are several expressions in the literature to define the equilibrium constant as a function of temperature [5-7]. These relationships are in good agreement with each other. In the present work, the formula given by Egan et al. [6] has been adopted due to the fact that it covers a wide range of temperature:

$$\log K_p = -\frac{8.2291\times10^3}{T} + 23.969 \qquad (6)$$

In order to further improve the accuracy of the model, in addition to the data obtained from Egan et al. [6], existing equilibrium data at the industrial reactor outlet was also considered and Eq. (6) was improved as follows:

$$\log K_p = -\frac{7.6569\times10^3}{T} + 22.161 \qquad (7)$$

The formation of urea and biuret are considered as slow reactions and far from the equilibrium in the urea reactor. For the urea formation reaction, the rate equation of Claudel et al. [8] in the presence of initial water was used in the present study:

$$\frac{dX_1}{dt} = k(1-X_1)(a+X_1) \qquad (8)$$

In the case of biuret formation reaction, the rate equation proposed by Shen [9] was used in this study:

$$\frac{dy}{dt} = k'(C_{U0} - y)^2 \qquad (9)$$

Reactor Model

A schematic diagram of an industrial urea reactor is shown in Figure 1.

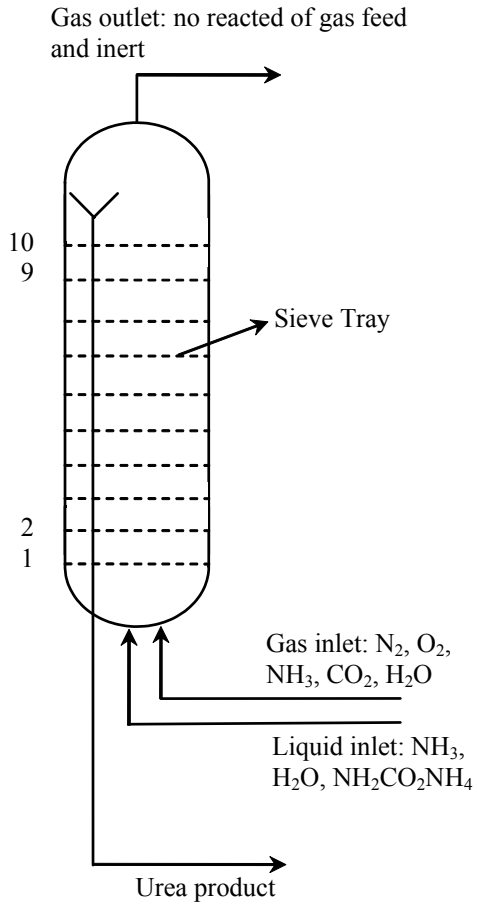


Figure 1: Urea reactor

In order to reach a model for the urea reactor the hydrodynamic and reaction submodels should be coupled. The urea reactor is a cylindrical vessel where nitrogen, oxygen, water, ammonia and carbon dioxide in gas phase and ammonium carbamate, free ammonia and water in liquid phase are introduced from the bottom. As it was mentioned above, in order to increase the mixing in the reactor, several perforated trays are installed inside the reactor. All the three main reactions (i.e., the heterogeneous reaction of formation of ammonium carbamate and urea and biuret formation in liquid phase) are considered to take place in the reactor. Irazoqui et al. [2] considered the whole urea reactor as a sequence of CSTRs. Although the same approach is adopted in this work for modeling the urea reactor, the heterogeneous reaction of ammonium carbamate formation (reaction 4) was adopted instead of the homogeneous reaction (reaction 2). A schematic model of sequential CSTRs is shown in Figure 2.

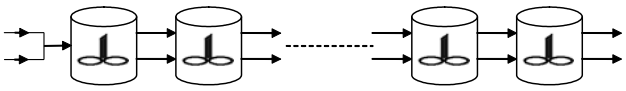


Figure 2. Schematic diagram of sequential CSTRs

Each reactor operates adiabatically. Due to high residence time of the reactants in the urea reactor, formation of ammonium carbamate at the reactor outlet can be practically considered at equilibrium [5]. However, it is obvious that the equilibrium is not reached in the interstages when dividing the reactor into *n* CSTRs. Therefore, it is assumed in the present study that the reaction proceeds only 1/*n* toward the equilibrium in each CSTR. As a result, the reaction would reach the equilibrium at the exit of the *n*th CSTR which is the outlet of the main urea reactor. The goal of this assumption is to ease the material and energy balance around each CSTR from which the temperature and compositions of outlet flows are calculated. Therefore, the amount of ammonium carbamate, urea and biuret could be calculated and then heat balance was performed for each CSTR. When the heat balance around a CSTR is satisfied, the calculated temperature and compositions of its outlet stream is used as the input data of the inlet stream for the next reactor.

The temperature of each CSTR has to be obtained through a trial and error method. This method is described as follows: The exit temperature of the reactor is guessed at the beginning of the calculations. Knowing the temperature, the equilibrium constant was calculated from eq. (8) and the amount of

ammonium carbamate formation is calculated from:

$$K_p = P_{NH_3}^2 P_{CO_2} = \left(\frac{n-2x}{n_t}\right)^2 \left(\frac{m-x}{n_t}\right) P^3 \quad (10)$$

Since it is assumed that in each CSTR this reaction moves 1/*n* toward the equilibrium, the amount of ammonium carbamate calculated from eq. (10) was multiplied to 1/*n*. Consequently, the amount of heat produced in the reactor due to ammonium carbamate formation was evaluated.

In the case of kinetic-controlled reactions (reactions 3 and 5) the corresponding conversion has to be determined from the mass balance equation of the CSTR [10]:

$$\frac{V}{v_0} = \frac{C_{A_0} X_A}{-r_A} \quad (11)$$

Introducing the kinetic expression of urea formation (eq. 8) into eq. (11) yields the following equation from which the amount of urea produced in each CSTR was calculated:

$$X_1^2 + \left(a + \frac{v_0}{kV} - 1\right) X_1 - a = 0 \quad (12)$$

Similar to the ammonium carbamate reaction, the amount of heat consumed due to urea formation could be calculated.

Biuret is also formed in the urea reactor. Introducing the reaction rate of biuret formation (eq. 9) into eq. (11) allows the amount of biuret at each CSTR to be calculated:

$$X_2^2 - \left(2 + \frac{v_0}{k'VC_{U0}}\right) X_2 + 1 = 0 \quad (13)$$

This is also an endothermic reaction whose required heat of reaction was evaluated after determining the amount of biuret formed in each CSTR.

Finally, for each CSTR the energy balance should be satisfied. The steady-state heat balance equation for each reactor is:

$$H_I - H_O + Q_{ac} - Q_u - Q_b = 0 \quad (14)$$

If eq. (14) is not satisfied, the assumed temperature of the reactor outlet was changed accordingly and the above mentioned calculations were repeated from the beginning. The iterative procedure was repeated until the energy balance equation is satisfied from which the exit temperature and composition of each CSTR were established.

Results and Discussion

Typical industrial operating conditions of the urea reactor are listed in Table 2.

Stream	P (kg/cm^2)	T (ºC)	Carbon Dioxide (kg/hr)	Ammonia (kg/hr)	Water (kg/hr)	Nitrogen (kg/hr)	Oxygen (kg/hr)	Ammonium Carbamate (kg/hr)
Gas	146	169.5	17505	30225	641	1443	205	0
Liquid	146	169.5	0	18924	14314	0	0	128220

Table 2. Typical inlet conditions of the reactor

Before starting the simulation, the number of stages (CSTRs) in the urea reactor has to be determined. Figure 3 shows the impact of the number of stages on the predicted conversion at the outlet of the urea reactor for the given set of operating conditions. The figure shows that beyond 10 stages, there is no significant change in the performance of the reactor. Therefore, in the rest of this work, the urea reactor was considered to be consisted of 10 CSTRs in series. It is worth noting that this number is also the number of sieve trays used in the industrial reactor. In each tray, the gas and liquid passing through the reactor mix again with each other and re-distribute the concentrations and temperature. Therefore, also from this point of view, each tray could be considered as a single CSTR. It is worth mentioning that although all the reactions considered by Irazoqui et al. [2] were in liquid phase, they presented a diagram very similar to what shown in this work in Figure 3. Irazoqui et al. [2] preferred to choose 25 stages for modeling the urea reactor. However, it is seen from Figure

3 that there would be no significant difference between these choices for the number of CSTRs.

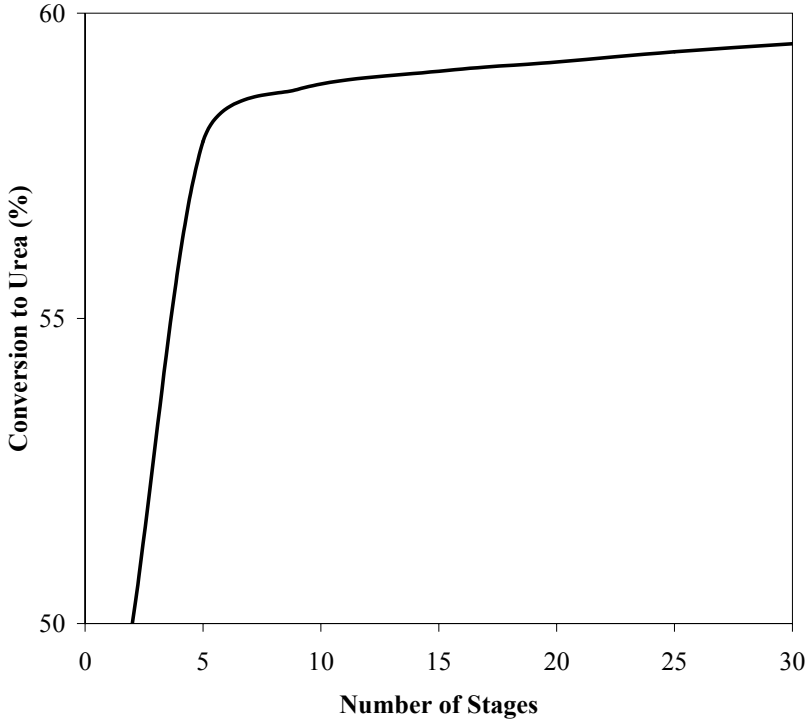


Figure 3. Impact of the number of CSTRs on the model prediction of the conversion to urea

The validity of the proposed model was tested against the data obtained from industrial scale reactor. Figure 4 illustrates the comparison between the predicted temperature profile along the reactor and real plant data. As can be seen in this figure, the model is in good agreement with the actual plant data. Moreover, a comparison between the model predictions and the plant data on some key operating parameters of the reactor is shown in Table 3. This table also confirms that the model predictions are in good agreement with the plant data.

Key parameters	Unit	Plant data	Simulation results
T_f	ºC	183	182.5
N/C	-	2.9	3.09
X_u	Wt%	33.9	33

Table 3. Comparison between plant data and modelling results for the urea reactor

Conclusions

The industrial urea reactor was modelled using a sequential modular approach. In order to develop this model, hydrodynamic and reaction submodels were coupled with each other. The hydrodynamic of the urea reactor was simulated by a sequence of CSTRs in series. The heterogeneous reaction of the formation of ammonium carbamate was considered in the model. Temperature dependence expression of equilibrium constant of ammonium carbamate reaction was corrected using the data in the literature as well as that at the exit of the real urea reactor. Comparison between simulation results and plant data shows a good consistency between the model and reality.

Acknowledgment

The authors are grateful to Khorasan Petrochemical Company for supporting this work.

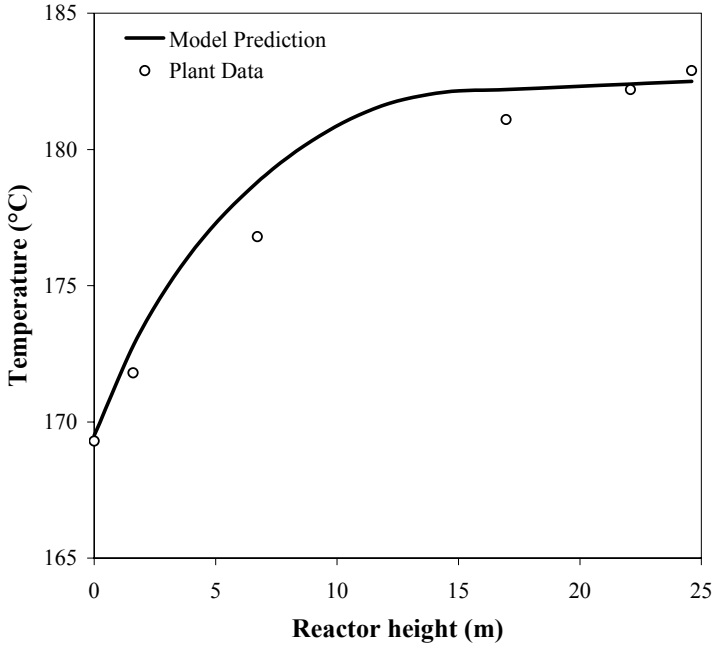


Figure 4. Comparison of the predicted temperature profile with actual plant data

References

[1] Isla M. A., Irazoqui H. A. and Genoud C. M. *Simulation of a Urea Synthesis Reactor. 1.Thermodynamic Framework*. Ind. Eng. Chem. Res. 1993; 32: 2662-2670.

[2] Irazoqui H. A., Isla M. A. and Genoud C. M. *Simulation of a Urea Synthesis Reactor. 2. Reactor Model*. Ind. Eng. Chem. Res. 1993; 32:2671-2680.

[3] Dente M., Pierucci S., Sogaro A., Carloni G. and Rigolli E. *Simulation Program for Urea Plants.* Comput. Chem. Engng. 1988; 21(5):389-400.

[4] Satyro M. A., Li Y., Agarwal R. K. and Santollani O. J. Modeling Urea Processes. *A New Thermodynamic Model and Software Integration Paradigm*. From the Virtual Materials Group. Presented at the Chemical Engineers' Resource Page. Hppt://www.virtualmaterials.com

[5] Joncich M. J., Solka B. H. and Bower J. E. *The Thermodynamic Properties of Ammonium Carbamate (An experiment in heterogeneous equilibrium).* J. Chem. Edu. 1967; 44(10):598-600.

[6] Egan E. P., Potts J. E. and Potts G. D. *Dissociation Pressure of Ammonium Carbamate.* Ind. and Engi. Chem. 1946; 38(4):454-456.

[7] Janjic D. *Etude de I'equilibre de dissociation du carbamate d'ammonium a I'aide d'un appareil nouveau*. Helvetica Chimica Acta 1964; 47(206) :1879-1883.

[8] Claudel B., Brousse E. and Shehadeh G. *Novel Thermodynamics and Kinetics Investigation of Ammonium Carbamate Decomposition Into Urea and Water*. Thermochemica Acta 1986; 102:357-371.

[9] Shen, R.C. *Rate of Biuret Formation from Urea.* J. Agr. Food Chem. 1959; 7(11):762-763.

[10] Fogler H.S. *Elements of chemical reaction engineering*, 3rd ed., Englewood Clifts, Prentice-Hall, 1986.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

MODELLING THE INFLUENCE OF THE GAS TO MELT RATIO ON THE FRACTION SOLID OF THE SURFACE IN SPRAY FORMED BILLETS

J.H. Hattel[1] and N.H. Pryds[2]

Process Modelling Group
Department of Manufacturing Engineering and Management
Technical University of Denmark, Building 425
DK-2800, Lyngby Denmark

[2]Materials Research Department, Risø National Laboratory, DK-4000 Roskilde, Denmark

Abstract

In this paper, the relationship between the Gas to Melt Ratio (GMR) and the fraction solid of an evolving billet surface is investigated numerically. The basis for the analysis is a recently developed integrated procedure for modelling the entire spray forming process. This model includes the atomisation stage taking thermal coupling into consideration and the deposition of the droplets at the surface of the billet taking geometrical aspects such as shading into account. The coupling between these two models is accomplished by ensuring that the total droplet size distribution of the spray is the summation of "local" droplet size distributions along the r-axis of the spray cone. The criterion for a successful process has been a predefined process window characterised by a desired fraction solid range at a certain distance from the atomizer. Inside this process window, the gas and melt flows have been varied and their influence on the fraction solid at the surface of the billet has been analysed.
Keywords: Spray forming, atomization, deposition, Modelling, Gas to Melt Ratio, Process window.

Nomenclature

A_g	Area of gas delivery nozzles [m]
c_p	Specific heat capacity [$JKg^{-1}K^{-1}$]
c_p^*	Adjusted specific heat capacity [$JKg^{-1}K^{-1}$]
d	Droplet diameter [m]
$d_{50,k}$	Mean droplet diameter of local distribution k [m]
$\boldsymbol{e}_n$	Unit vector normal to the preform [m]
$\boldsymbol{e}_f$	Unit vector along the line between surface point and the atomiser [m]
$\boldsymbol{e}_x$	Unit basis vector [m]
$\boldsymbol{e}_y$	Unit basis vector [m]
$\boldsymbol{e}_z$	Unit basis vector [m]
d_m	Mass median diameter of droplets [μm]
d_0	Liquid stream diameter [μm]
D_0	Distance from the atomizer to the surface of the billet [m]
f_s	Fraction solid [-]
f_l	Fraction liquid of spray [-]

F_l Fraction liquid of preform [-]
i Index for droplet diameter [-]
J_g Mass flow rate of gas [kgs^{-1}]
J_l Mass flow rate of liquid [kgs^{-1}]
k Thermal conductivity [Wm^{-1}K^{-1}]
k Index for local distribution [-]
K_a Experimental constant in (1) [m]
$\dot{m}$ Mass flow of droplets in diameter interval [kgs^{-1}]
$\dot{M}(r,z)$ Mass flux distribution in spray cone [kgs^{-1}]
r Spatial coordinate [m]
SE Sticking efficiency [-]
t Time [s]
T Temperature [K]
T_{sol} Solidus Temperature [K]
T_{liq} LiquidusTemperature [K]
T_M Solvent melt Temperature [K]
T_E Eutectic Temperature [K]
P_k^{local} Local droplet distribution [-]
We Weber number [-]
x Spatial coordinate [m]
y Spatial coordinate [m]
z Spatial coordinate [m]

α_s Parameter in (4) [-]
α_l Parameter in (4) [-]
ΔH_f^{Left} Remaining freezing enthalpy [Jkg^{-1}]
ε Parameter in (4) [-]
η_g Kinematic viscosity of gas [Pa s]
η_l Kinematic viscosity of liquid [Pa s]
ρ Density [kgm^{-3}]
σ Standard deviation [-]
θ Spatial coordinate [-]
ξ Shading function [-]

Introduction

Despite a continuous attempt towards fundamental understanding of the spray forming process, many important features remain unexplored. An enhanced understanding of the process should permit development of predictive models elucidating the interrelationships between the process/structure/properties and performance.

The spray forming process has been continuously modelled and described in literature. The models are often divided into two parts namely: atomization and deposition. A major part of the models for atomization are based on the idea that the continuous phase affects the properties of the dispersed phase but not vice versa [1,2]. However, in a real system the droplets, which are represented by their size distribution do interact with each other via the gas and therefore a model that is able to reflect that, is desirable.

As for the deposition, the models proposed in literature can be divided into two principally different approaches, i.e purely geometrical models [3-4] and models, which take both the thermal and the geometrical effect into consideration [5].

Recently, a new integrated model for both atomization and deposition in the spray forming process taking into account the interaction between the atomization gas and the different droplet sizes has been developed for a Gaussian shape [6-8] and billet shape [9] geometries.

One of the important process parameters for controlling the quality of the spray formed billet in the industry is the surface fraction solid. This is very much dependent on the energy contained in the droplets arriving to the surface of the preform, i.e. the temperature and size of the droplets. Moreover, the droplet size distribution as well as the temperature of the droplets are dependent on the ratio between the gas flow and the melt flow and on the gas velocity. A way to control the surface fraction solid during the process, is to control the ratio between the flow rate of the gas and the melt known as the Gas to Metal Ratio (GMR).

In the present work, the recent developed numerical model for the spray forming process [9] has been used to study the influence of the GMR on the surface fraction solid.

Model description

Atomization model

During atomization, a bulk liquid is disintegrated into droplets of different sizes. Looking at a given atomizer design and melt composition, several variables influence the droplet size distribution which in turn is decisive for the thermal state of the arriving droplets at the preform and hence the surface temperature. These variables include process related variables such as: Melt superheat,

nozzle geometry, melt flow rate, gas flow rate, gas velocity, gas temperature, and chemical and physical properties of the liquid metal and the atomization gas, [1].

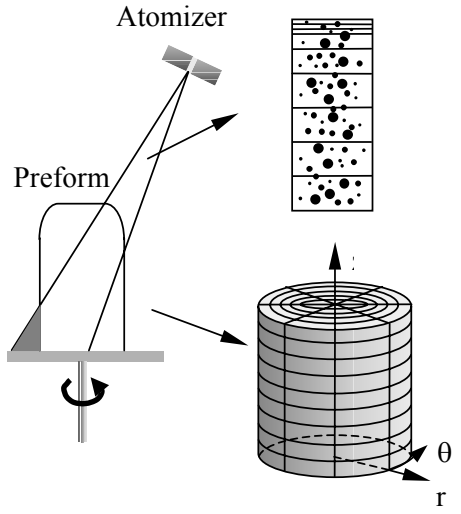


Figure 1 Integrated model of the spray forming process consisting of a 1-D model of the atomization in the spray cone and a 3-D model of the deposition process.

In order to solve correctly the two-phase flow problem, i.e. the gas phase and the liquid metal phase, during the atomisation process, a fully coupled momentum and energy formulation should be applied in general. In the present model, the momentum and energy conservation of the droplets during flight are coupled together. The droplet size distribution is discretized by a number of size groups ranging from 1 up to 600 μm in diameter with a constant increment of Δd. The upper limit is chosen as the maximal possible size of the droplets in the distribution. Each group's contribution to the overall heat balance is given by its mass flow, i.e. its probability times the total mass flow of metal, J_l, see equation (3).
The heat balance between the droplets and the surrounding gas is set up by assuming a 1-D Eulerian frame, i.e. fixed finite control volumes along the centreline of the spray cone, assuming that the injected gas is only slightly expanded along the radial direction. This assumption enables to simulate the process in one dimension, see Figure 1. It should be noted that thermal coupling was present in the atomization model, because of the fact that the gas temperature was not assumed to be known a priori, but calculated together with the droplet temperatures in a coupled manner. A more comprehensive description of the atomization model is given in [6,8].

The mean diameter of the droplet size distribution is directly related to the GMR and several suggestions for this relationship have been given in literature. In the present model, the most comment used empirical equation representing the mass median diameter has been used, [10]:

$$\frac{d_m}{d_0} = K_a \left[\frac{\eta_l}{\eta_g We} \left(1 + \frac{J_l}{J_g} \right) \right]^{0.5} \quad (1)$$

From this expression it is readily seen that increasing GMR decreases the mass median diameter and thereby the droplet size. The temperature of the arriving particles at the deposit surface depend both on the particle size, i.e. smaller droplets cool faster than large ones, as well as on the position of the deposit substrate along the axis of symmetry. This leads to an enthalpy of the spray cone, which is position dependent and influences the surface fraction solid of the billet.
In order to achieve a good homogeneous and dense billet, the desirable fraction solid of the particles on impact should be around 0.5-0.7 [1]. This criterion defines the desired process window used later on in the analysis.

Deposition model
The *deposition model* is based on a 3-D cylindrical geometry, (r,θ,z). The temperatures in the spray deposit material are thus governed by

$$\rho c_p \frac{\partial T}{\partial t} = \frac{\partial}{\partial r}\left\{ k \frac{\partial T}{\partial r} \right\} + \frac{1}{r} k \frac{\partial T}{\partial r} + \frac{1}{r^2} \frac{\partial}{\partial \theta}\left\{ k \frac{\partial T}{\partial \theta} \right\} + \frac{\partial}{\partial z}\left\{ k \frac{\partial T}{\partial z} \right\} + Q_{gen} \quad (2)$$

In the present case Q_{gen} arises from the phase change. Note that the release of the *remaining latent heat* after droplet flight, ΔH_f^{Left}, in the preform, is accomplished by introducing an adjusted value for the specific heat, c_p^*, according to the expression

$c_p^* = c_p - \Delta H_f^{Left} \frac{\partial f_s}{\partial T}$, where the term $\frac{\partial f_s}{\partial T}$ is given by Clyne-Kurz model. The derivation of c_p^* is easily obtained by setting Q_{gen} equal to the released heat per unit volume from solidification in equation (2). Combining this temperature dependent c_p^*-value with an algorithm "pushing" the temperature back to the liquidus temperature when entering the solidification interval [11], no iterations are needed for this non-linearity in order to ensure consistency between temperatures and material data.
The numerical solution of equation (2) is based on the finite volume method, in which the governing differential equation is integrated over each of the control volumes. This way, the conservation principle, i.e. energy balance is expressed for the control volume. Further details on the choice of boundary conditions and heat transfer coefficients are given elsewhere, [8].

Coupling of the models
The coupling of the atomization and the deposition model is achieved through the droplet size distribution. The total size distribution of the spray is in fact the summation of a number of "local" distributions along the r-axis. The mesh along the r-axis is divided into control volumes each of which contains a droplet size distribution, $P_k^{local}(d_i)$, i.e. a "local" distribution. The overall mass distribution can be expressed as,

$$\begin{aligned} &\dot{m}(d \in [d_i - \tfrac{\Delta d}{2}; d_i + \tfrac{\Delta d}{2}]) = \\ &J_l \sum_k \int_{\ln(d_i - \frac{\Delta d}{2})}^{\ln(d_i + \frac{\Delta d}{2})} P_k^{local}(d_i) d(\ln d_i) \approx \\ &J_l \sum_k P_k^{local}(d_i) \Delta(\ln d_i) \\ &= J_l \sum_k \frac{1}{\sigma_k \sqrt{2\pi}} \exp\left(\frac{-(\log d_i - \log d_{50,k})^2}{2\sigma_{kj}^2} \right) \\ &(\log(d_i + \tfrac{1}{2}\Delta d) - \log(d_i - \tfrac{1}{2}\Delta d)) \end{aligned} \quad (3)$$

where the local dispersion is assumed to depend on the local mass mean similarly to the global distribution, i.e. $\sigma_k = \sqrt[3]{\frac{d_{50,k}}{13}}$, [12].
The local size distribution, i.e. in each control volume, is then used to calculate the average enthalpy of each control volume. The major advantage of the atomization model is that it enables a calculation of the enthalpy for each group of similar droplet size, based on the interaction of this size group with the whole range of sizes and the surrounding gas. The enthalpy of the different droplet sizes is then contributing to the overall enthalpy of the specific control volume. Subsequently, the enthalpy is converted into temperature and used to describe the temperature of the specific cell. For more details on this procedure see [9].

A key parameter, which determines the yield and the shape of the deposited material, is the sticking efficiency, SE of the droplets to the surface. The sticking efficiency describes how good the droplets "stick" to the already deposited material. The model proposed by Mathur [13] was applied for the numerical procedure. This SE model is based on two macroscopic components: a) A geometrical component which depends on the angle of incident between the spray direction and the surface normal and b) A thermal component which depends upon the fraction liquid in the spray and the fraction liquid in the deposited surface. The thermal SE(T) is given by,

$$SE(T) = (f_s \alpha_s + f_l \alpha_L)\varepsilon \quad (4)$$

where

$\alpha_s = 1 - 0.75 \cdot (1 - F_L)$ and $\alpha_l \approx 0.98$

and the parameter ε varies from 0 to 1 and is consistent with the variation in the viscosity as a function of F_L.

Billet Shape model
The spray forming process in the industry is used mainly for the production of billet shape products. The final billet shape and its properties are strongly influenced by the process parameters and therefore the ability to control the final shape is very important.
The billet model in the present work describes the evolution of a 3D deposit surface in Cartesian coordinates. The important effect of shading has to be incorporated into the model in order to simulate the correct shape of the billet, see Fig. 1. Previous studies [14] have already taken this effect into consideration, however, in the present study a somewhat different

approach is used to calculate the shading effect [15].
The billet shape model is initiated by assigning a number of grid points to the substrate surface and thereby defining a mesh, composed of triangles. The position of these grid points is then advanced with time. In each time step of the calculation, the new position of all the points on the surface is calculated using the mass flow distribution and geometrical considerations. The new position is found as [9], see Figure 2:

$$\begin{aligned} x_{new} &= x + \Delta h \cdot \mathbf{e}_n \cdot \mathbf{e}_x \\ y_{new} &= y + \Delta h \cdot \mathbf{e}_n \cdot \mathbf{e}_y \\ z_{new} &= z + \Delta h \cdot \mathbf{e}_n \cdot \mathbf{e}_z \end{aligned} \quad (5)$$

and

$$\Delta h = \int_t^{t+dt} \xi(x,y,x,t) \cdot \dot{M}(r,z) \cdot \mathbf{e}_n \cdot \mathbf{e}_f \, dt \quad (6)$$

$\dot{M}(r,z)$ is the mass flux and ξ(x,y,z,t) is the shading function which is designed in such a way that ξ is equal to 0 when shading is present, otherwise it is 1.

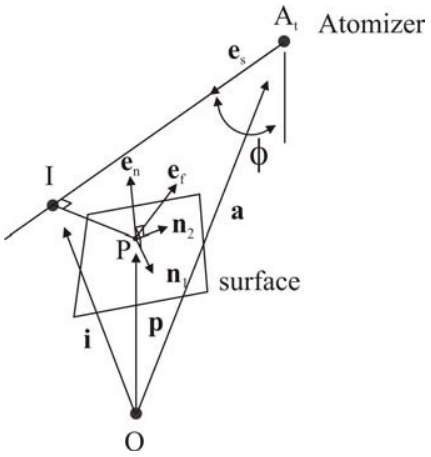


Figure 2 Vectors describing the evolution of the billet surface.

In order to calculate the surface evolution, i.e. Δh, the four terms, $\mathbf{e}_f$, $\mathbf{e}_n$, $\dot{M}$ and ξ must be obtained. The vectors $\mathbf{e}_f$ and $\mathbf{e}_n$ are obtained from relatively straightforward vector calculus, see e.g. [8, 9]. The mass flow at any point at the surface of the preform can be found from the mass distribution in the spray cone, e.g. typically given by a Guassian distribution in combination with the actual positions of the atomizer and the point under consideration, [9]. The determination of the shading function, ξ, calls for a rather complex algorithm. The details of the different sub-models and validations against analytical and numerical solutions as well as experimental observations have been extensively published and can be found elsewhere, e.g. [8,9].

Results

Figures 3 and 4 show a comparison between the shape calculated by the present model and experimentally produced billets of 100Cr6 steel. The material data used for these calculations as well as the properties of the atomization gas are given in Table 1-2.

Table 1
Material properties for 100Cr16.

Solidus temperature T_{sol} [K]	1570
Liquidus temperature T_{liq} [K]	1724
Solvent melt temperature T_M [K]	1811
Eutectic temperature T_E [K]	1419
Density ρ [kg/m^3]	7810
Specific heat of solid C_p^{sol} [J/KgK]	640
Specific heat of solid C_p^{liq} [J/KgK]	724

Table 2
Properties of the atomization gas, N_2.

Density ρ [kg/m^3]	1.5
Viscosity η_g [Ns/m^2]	0.000032
Specific heat C_p [J/kgK]	1000
Thermal conductivity k [W/mK]	0.026

Table 3
Process parameters used for the calculation

Mass distribution parameter b [m^{-2}]	1000
Rotation velocity ω [rot/min]*	116
Spray angle ϕ [deg]	35
Excentric distance l_e [m]	0.02
Distance to atomizer D_0 [m]	0.5
Withdrawal velocity V	**
Constant in Eq. (1), K_a	50

* The rotation velocity is chosen to be 116 rot/min since literature suggests that this should be high enough to ensure rotational symmetry, [6].

** The withdrawal velocity is chosen in such a way that the distance from the center of the billet to the atomizer at all times is D_0 in order to keep the solid fraction of the droplets arriving to the surface of the deposit constant.

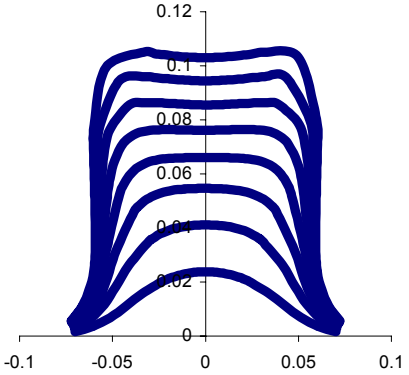


Figure 3 Prediction of billet shape of 100Cr6.

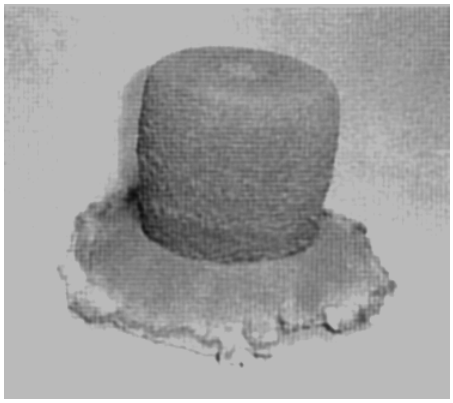


Figure 4 Experimentally obtained billet shape of 100Cr6.

The geometrical process parameters are given in Table 3.

As seen from Fig. 3, the model is able to predict the 'neck' at the beginning of the deposition as also observed in real billets. Such a 'neck' cannot be predicted without taking the shading effect into consideration. This is a direct indication of the effectiveness of the shading procedure.

For the purpose of analyzing the influence of GMR on the surface solid fraction of the billet, two process parameters have been varied, i.e. melt stream diameter and the area of gas delivery nozzles, see Table 4.

The basic atomizer geometry was chosen according to a melt stream diameter of d_0 = 3700 μm and an area of gas delivery nozzles of A = 0.0003 m^2, see Table 4, cases 1, 2.1 and 3.1.

In order to achieve more comparable results, the atomizer geometry (melt stream diameter and nozzle area) was changed in such a way that the GMR and the gas velocity at the exit were constant as compared to case 1 (see cases 2.2 and 3.2). The gas velocity is given by the expression, [10]

$$v_g = \frac{J_g}{\rho_g A_g} \exp(-z/\lambda_v) \qquad (7)$$

where λ_v is the decay constant [10].

As seen from equation (7), the gas velocity is inversely proportional to nozzle area. Thus, using a metal flow rate of 0.05 kgs^{-1} and a nozzle area of 0.0003 m^2 as reference, i.e. case 1 (and keeping in mind that GMR was chosen as 1), flow rates of 0.1 and 0.2 kgs^{-1} result in a nozzle area of 0.0006 and 0.0012 m^2 in case 2.2 and 3.2, respectively. From equation (7) it is also seen that enforcing a constant gas velocity results in the gas area being proportional to the gas flow. Hence, the melt stream diameters were then adjusted as 3700, 5232 and 7400μm for metal flow rates of 0.05, 0.1 and 0.2 kgs^{-1}, respectively. See Table 4.

Figures 5-7 show the calculated solid fraction as a function of distance from the atomizer. In these figures, the four vertical and horizontal lines correspond to the distance from the atomiser and the solid fraction of the spray, respectively. These lines represent the limit values of the process parameters. The fraction solid is marked with two lines, 0.45 is set as the lower limit and 0.75 is set as the upper limit. The distance from the atomizer in which the substrate is situated is also marked with two lines – one at a distance of 0.5m and one at a distance of 0.6m - typical distances used in the industry [16].

As seen from these figures, increasing the mass rate causes increase in the fraction solid. If the atomizer geometry is kept

constant, i.e. without modifying the nozzle dimensions, the fraction solid of the spray for a mass flow of 0.1 kg/s is above the desirable fraction solid for GMR=1.5 and GMR=2.0. In the case of a mass flow of 0.2 kg/s, the fraction solid already exceeds the limits at a GMR=1.0 and for a GMR=1.5 it is far above the limit and approaching a fraction solid of 1.

Table 4 Varying process parameters for the five different calculation cases in Figs. 5-10.

Calculation case	1
Mass flow J_g	0.05
Change in atomizer geometry	No
Melt stream diameter D_0	3700
Area of gas delivery nozzles A	0.0003

Table 4 continued

2.1	2.2	3.1	3.2
0.1	0.1	0.2	0.2
No	Yes	No	Yes
3700	5232	3700	7400
0.0003	0.0006	0.0003	0.0012

Figures 8 – 10 show the fraction solid as a function of the distance from the atomizer for different values of the GMR and constant mass flow rates of 0.05, 0.1 and 0.2 kg/s, respectively. For the two latter, the modified nozzle data were used, i.e. cases 2.2 and 3.2 in Table 4. The calculations were made in order to evaluate which values of the GMR fit the process window. As seen in Figure 8 for a mass flow of 0.05kg/s and a GMR varying from 1 to 2, the fraction solid of the spray is within the desirable range, whereas for a mass flow of 0.1 kg/s, the fraction solid is in the desirable range for GMR=1-2.5 (Figure 9). For a mass flow of 0.2kg/s the range of GMR is varying from 1.25 to 3 (Figure 10).
It is seen from Figures 8 – 10, that the range of acceptable GMRs, in which the solid fraction has the desirable value, i.e. 0.45-0.75, is larger with an increasing mass flow.

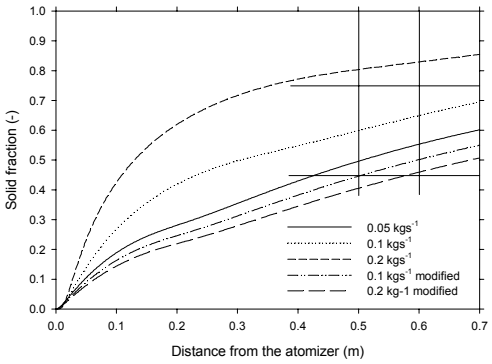


Figure 5 Average fraction solid of the spray as function of distance from atomizer and mass flow. GMR=1.

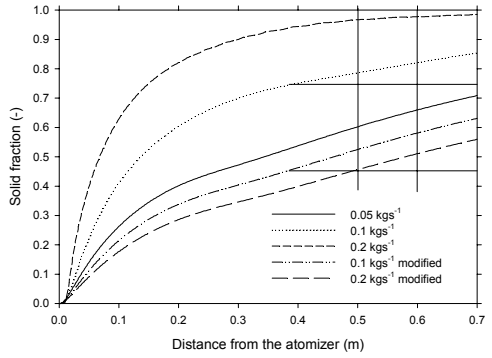


Figure 6 Average fraction solid of the spray as function of distance from atomizer and mass flow. GMR=1.5.

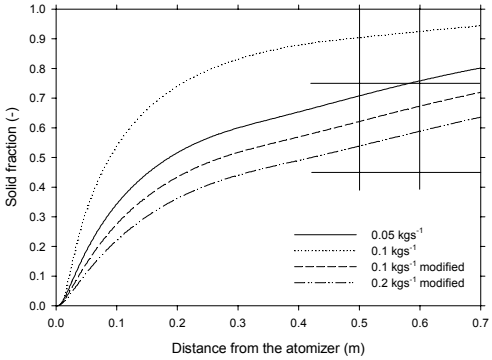


Figure 7 Average fraction solid of the spray as function of distance from atomizer and mass flow. GMR=2.

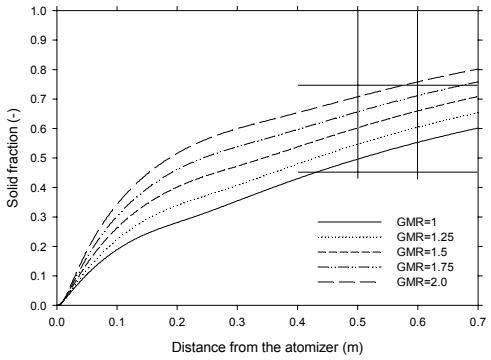


Figure 8 Average fraction solid of the spray as function of distance from atomizer and GMR. Mass flow of 0.05kg/s.

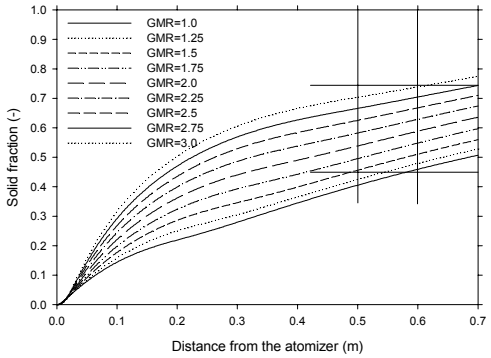


Figure 9 Average fraction solid of the spray as function of distance from atomizer and GMR. Mass flow of 0.1kg/s. Modified nozzle data, calculation case 2.2.

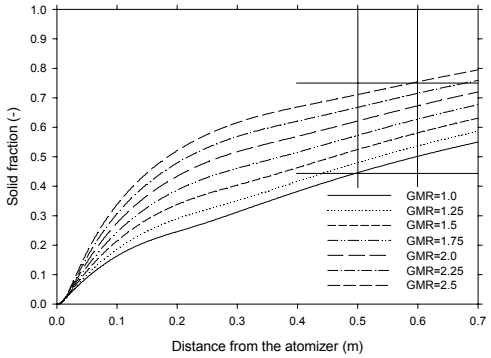


Figure 10 Average fraction solid of the spray as function of distance from atomizer and GMR. Mass flow of 0.2kg/s. Modified nozzle data, calculation case 3.2.

Conclusion

In the present work, the relationship between the gas to melt ratio (GMR) and the surface fraction solid of an evolving billet surface in the spray forming process has been investigated numerically. This has been done with an integrated approach, which models the entire process taking both atomisation and deposition into account. With a predefined process window, the model was used to analyse the influence of changing the nozzle geometry on the relationship between GMR and the surface fraction solid at a certain distance from the atomizer. This way it was shown that the present model could be used to evaluate different process parameters, such as nozzle geometry, for the purpose of doing process optimization.

References

[1] Y. Zhou, Y. Wu and E. J. Lavernia, "Process modeling in spray deposition: a review", International J. Non-Equilibrium Processing, 10, (1997), 95-183

[2] C. T. Crowe, M. Sommerfeld and Y Tsuji, Multiphase Flows with Droplets and Particles, CRC Press, 1997

[3] I.A. Frigaard, SIAM Journal on Applied Mathematics., 57 (3), p. 649-682, 1997

[4] I.A. Frigaard, Journal of Materials Processings and Manufacturing Science., 3, p. 173-193, 1994

[5] P. C. Mathur, S. Annavarapu. D. Appelian and A. Lawley, Materials Science and Engineering A142, 261-276, 1991

[6] J.H. Hattel, N.H. Pryds, J. Thorborg and P. Ottosen, "A quasi-stationary numerical model of atomized metal droplets. I: Model formulation", Modelling Simul. Mater. Sci. Eng., 7, (3), (1999) 413-430

[7] N.H. Pryds, J.H. Hattel and J. Thorborg, , "A quasi-stationary numerical model of atomized metal

droplets. II: Prediction and Assessment", Modelling Simul. Mater. Sci. Eng., 7, (3), (1999) 431-446

[8] N.H. Pryds, J.H. Hattel, T.B. Pedersen and J. Thorborg, "An Integrated Numerical Model of the Spray Forming Process", Acta Materiala, 50, (16), (2002), 4075-4091

[9] J.H. Hattel and N.H. Pryds,, "A Unified Spray Forming Model for the Prediction of Billet Shape Geometry, Acta Materiala, (To be published)

[10] P. S. Grant, B. Cantor and L. Katgerman, Acta Metall. Mater., 41(11), (1993), 3097

[11] J.Hattel (ed.), "Fundamentals of Modelling of Casting Processes", Nova Science (to be published)

[12] H. Lubanska, Journal of Metals, 22 (1), (1970), 45-49

[13] P. Mathur, D. Apelian and A. Lawley, Acta Met. 37, 429, 1989

[14] H.-K. Seok, H. C. Lee, K. H. Oh, J.-C. Lee, H.-I. Lee, and H. Y. Ra, Metallurgical and Materials Transactions A. 31A, 1479-1488, 2000

[15] J.Hattel and N.H. Pryds, "An Integrated Numerical Model for the Prediction of Gaussian and Billet Shapes", Mat.Sci.Eng. (In press)

[16] P.Kjeldsteen, Personal communication, Danspray, 2002.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

PREDICTION OF COATED PAPER PROPERTIES BY ARTIFICIAL NEURAL NETWORKS

Ales Hladnik*
Pulp and Paper Institute
1000 Ljubljana
Slovenia

Helmut Stark
Technical University Graz
Institute of Paper, Pulp and Fibre Technology
8010 Graz
Austria

Barbara Stravs-Grilc
Papermill Vevce B&B
1000 Ljubljana
Slovenia

Abstract

Computer-based artificial intelligence methods are gradually gaining more and more popularity also in the field of papermaking. Comparing to the conventional modelling methods, artificial neural networks (ANN) offer several advantages: they can handle complicate non-linear functions with large number of variables, do not require detailed knowledge about the system studied and are relatively easy to use. Various stages in the papermaking process, such as stock preparation, paper sheet formation, drying, coating and calendering affect quality of the end product in different ways. These relationships are often poorly understood and/or very complex in nature. In order to model characteristics of the coated wood-free paper produced in Slovenian paper mill B&B Papirnica Vevce, back-error propagation network models were developed. Results of the work are presented with a special consideration of practical implementation of the neural models in the paper mill environment.
Keywords: Papermaking, modelling, process control, artificial neural networks, paper properties

* Corresponding author. Phone: +386 1 2002 847, Fax: +386 1 4265 639, E-mail: ales.hladnik@icp-lj.si

Introduction

Modern industrial paper production is a typical example of a complex multivariate process. Technological parameters monitored during the individual production stages - stock preparation, paper sheet forming, drying, calendering, etc. - can be regarded as a set of interdependent variables with each of them describing one part of the overall system. Due to their inherent interrelations these variables must be observed simultaneously in order to extract meaningful information from them.

Prediction of behaviour, i.e. modelling, of such a complex system is not an easy task. In this study we developed several artificial neural network (ANN) models based on the process data from paper production in Slovenian paper mill Papirnica Vevce B&B in order to characterize final paper quality parameters, such as brightness, gloss, stiffness or curl.

Data that can be found in industrial reality generally suffer from severe limitations (Figure 1). Data are usually noisy, measurement results can be unreliable or even missing. Number of parameters characterizing individual technological operations is often very large and their interrelationships complicated. For example, during the individual technological steps of papermaking process, more than 100 variables are usually being monitored. The complexity of monitoring is further complicated by the fact that variables which individually show no specific influence on the quality of the end product, e.g. paper curl, may in combination with other parameters contribute significantly to this final paper property.

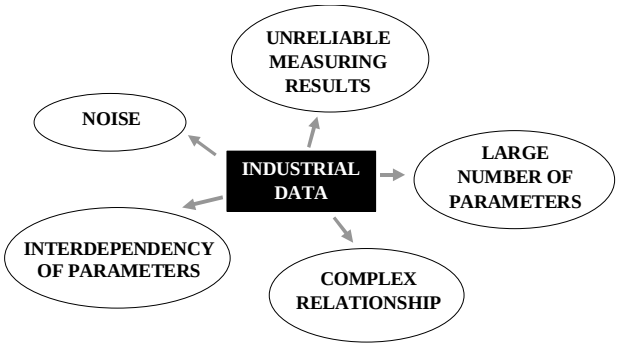


Figure 1: Nature of industrial data

Classical approach such as multiple linear regression would not be appropriate as a modelling tool for such a process. One of prerequisites for applying this tool is that the variables can be manipulated, i.e. varied independently from each other. In industrial environment this is usually not the case, since such trials would lead to the production of low-quality product or even to process down time. In addition, in modern manufacturing facilities there is often already a wealth of historic operating information, which is available for analysis.

Artificial neural networks

ANN allow user to build a model based on past experience (recorded data) rather than requiring a detailed knowledge of the system. Their major advantages over the more traditional modelling methods include handling of complex non-linear relationships among data, high processing speed and ease of use [1].

Of the many types of ANN, for modelling technical processes the most suitable ones are the so called feed-forward networks that apply back-error propagation (BEP) learning algorithm. This is the type of net that was also used in our research.

ANN consists of processing units called neurons or nodes [2]. Each neuron accepts one or more inputs (x_1 to x_d), multiplies each input by its associated connection strength (weight – w), sums the weighted inputs and uses an activation function (g) to produce its output. The neurons are arranged in interconnected layers. Figure 2 shows a three-layered network.

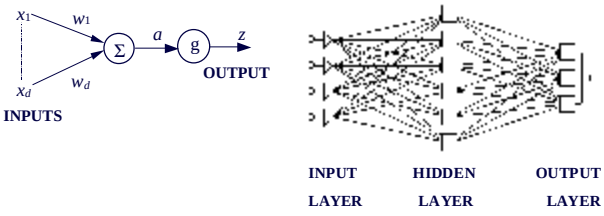


Figure 2: Artificial neuron (left) and three-layered network (right)

Training and testing

Central task of every NN is to produce a model that fits the actual experimental data (target outputs). In other words, network weights must be adjusted so that the prediction error made by the network is minimized. Modelling is a two-step

process (Figure 3). The historical cases that the experimentator has gathered are randomly divided into two groups: training and test set of samples. During training or learning, the network is presented with training samples of inputs together with the target, i.e. actual output(s) for each sample. The network processes the inputs for each sample to produce an estimated output, compares this value to the desired target and adjusts its weights to produce a more correct output. This process proceeds iteratively until the correct outputs are predicted by the NN model for both the training population and independently for a set of test samples not originally used for training. During this second, testing phase, the network's parameters are not changed anymore and the network is used for the reproduction of input data in order to "predict" suitable output data.

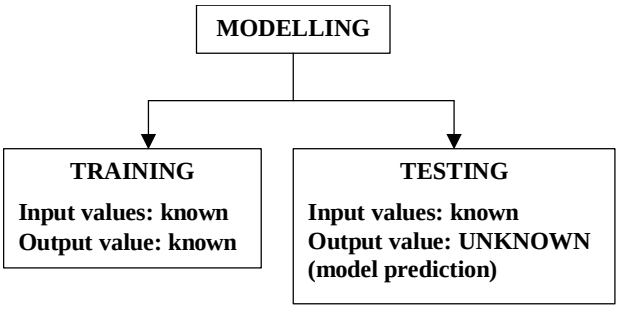


Figure 3. Modelling stages

Figure 4 schematically shows what happens during the training and testing phase of a neural modelling. While the training error monotonically decreases indicating a steady improvement in fitting to the actual training data, the test error exhibits a minimum which corresponds to an optimum network configuration and best prediction for a new, test set of samples. A further increase in the number of hidden neurons leads to an increase in test error, i.e. to overfitting.

Unfortunately, there is no general rule how to choose optimum number of hidden neurons, generalization and overfitting depend also on the number of training cases, the amount of noise and the complexity of the function one wants to model.

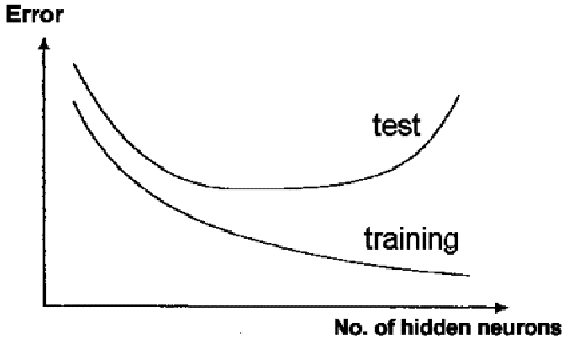


Figure 4: Network error as a function of hidden neurons number

Experimental

Database setup

The system we wanted to investigate in our study consists of a typical Fourdrinier paper machine followed by an off-line coater and calender. A single-sided coated wood-free paper for labels with a basis weight of 80 g/m^2 was chosen for modelling. Parameters monitored during the individual technological stages of paper production and surface treatment have been recorded and stored into the paper mill database PQM system. From altogether 121 process (on-line) and laboratory (off-line) variables that have constantly been measured during the production of this paper grade, some were excluded from the further analysis due to the missing or unreliable observations. Similarly, some of the samples, i.e. paper tambours were discarded for the same reasons. The remaining database consisted of 220 samples and 107 variables thus representing production history of these paper tambours.

Due to the partly confidential nature of this study, some details concerning paper production (e.g. range or magnitude of the monitored parameters) as well as data collection issues can not be revealed. For the same reason, variables names have been coded.

As already mentioned, one of the main goals of our work was to build neural models that would predict final quality of the paper under investigation based on data gathered during the paper production. Therefore we specified several – altogether eight – quality parameters of the finished, i.e. coated and calendered paper that have been regularly monitored using standard laboratory methods:

basis weight, calliper, smoothness, brightness, opacity, stiffness, gloss and curl. Each model contained only one such output – target – variable whose fluctuation was to be modelled. As inputs served all or part of the process parameters monitored during the production of a paper tambour: data on beating, chemicals used in the paper stock, numerous parameters from paper machine wet- end dry end, variables describing off-line coating and calendering process.

All inputs' model

The total database consisting of 220 samples was randomly divided into two groups: a larger one containing 164 training samples and a smaller one consisting of the remaining 56 samples which were used for testing the performance of the models.

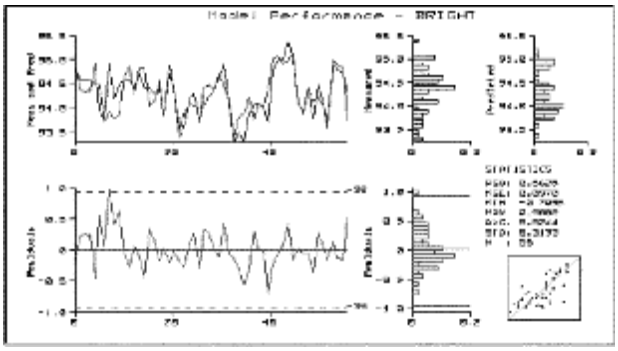


Figure 5: All-inputs' model for paper brightness

Summary of the testing performance of the neural network model for final paper brightness is displayed in Figure 5. In this model, all of the available process parameters – altogether 97 – were used as inputs to the network. In the upper left diagram we can see the degree of matching between the measured and the predicted brightness values for each of the 56 paper samples. Below, fluctuation of residuals, i.e. differences between the measured and the predicted values for each of these samples is displayed. The two biggest residuals, i.e. errors in neural net prediction, are 1.0 unit in positive (denoted as MAX = measured value is higher than the predicted) and 0.7 units in negative direction (MIN = predicted higher than measured). Squared correlation value (RSQ) of the model is 0.56.

Dimensionality reduction

It was pointed out earlier that several factors affect the prediction error of each particular network configuration. One of the most important factors determining ability of a particular neural model to successfully predict output variable is also the selection of input variables for that model. In literature [4] it has been reported that appropriate reduction of inputs, i.e. keeping only those having significant influence on the output variable should reduce the prediction error. Various strategies can be adopted, such as Minimal description length [5] or Bayesian model selection [6]. In practice, however, when dealing with a system of an extreme complexity such as paper production where numerous parameters are interrelated, this proves to be a very difficult task. We have therefore implemented several algorithms in an attempt to improve the predicting power of the neural models (Figure 6).

First we created lists of input variables according to their importance or impact to the particular final paper modelling property (e.g. brightness); those inputs having the biggest impact to the output were ranked at the top of these lists, those being the least connected were ranked at the end of the lists. We selected three different importance or impact criteria for each modelling property: inputs-output correlation coefficients, sensitivity analysis and combination of correlation coefficients and experts knowledge.

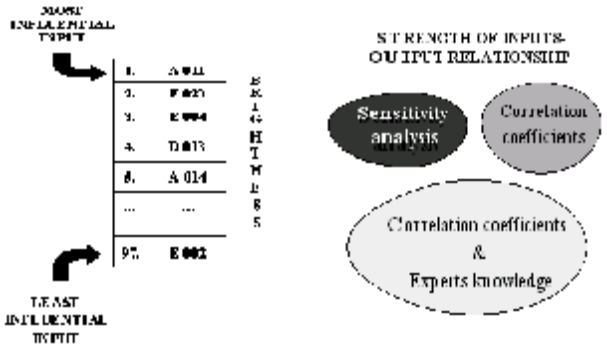


Figure 6. Ranking of influential inputs (left) based on various dimensionality reduction approaches (right)

When using the first method – correlation coefficients – inputs were ranked according to the magnitude of linear correlation to the final paper

property. Sensitivity analysis [7] reveals the sensitivity of the output variable to changes in the input variables; the most sensitive inputs were ranked at the top and the least sensitive ones at the bottom. Finally, we presented lists of inputs ranked according to the correlation coefficients criterion to the paper mill experts who made their suggestions and modified these lists. The three lists for each of the eight final paper modelling properties differed considerably with respect to the selection and importance of inputs for the particular modelling property, which supports the above mentioned statement about the difficulties of finding the optimum subset of inputs for modelling.

Models' performances

Next, we used 30 most influential variables from each of the list as inputs to the neural models. Together with the all-inputs mode we therefore created altogether 36 models, 4 for each of the 8 paper properties. Figure 7 shows a comparison among different inputs' selection approaches and their effect on model prediction power. The latter was estimated based on a correlation (Pearson correlation coefficient R) between the measured and the predicted values for respective modelling property for testing population: the higher the R value, the better the prediction of that final paper property. We see that in some cases – e.g. paper stiffness – reduction of inputs based on sensitivity analysis proved most successful, in others, such as with paper gloss – combination of correlation coefficients and experts knowledge led to the best results. Let's have a closer look at some of these models. Note that for each modelling property, the 30 variables' list with the highest R value served as inputs to that model, e.g. for basis weight this was sensitivity analysis, for caliper all (=97) input variables were retained, etc.

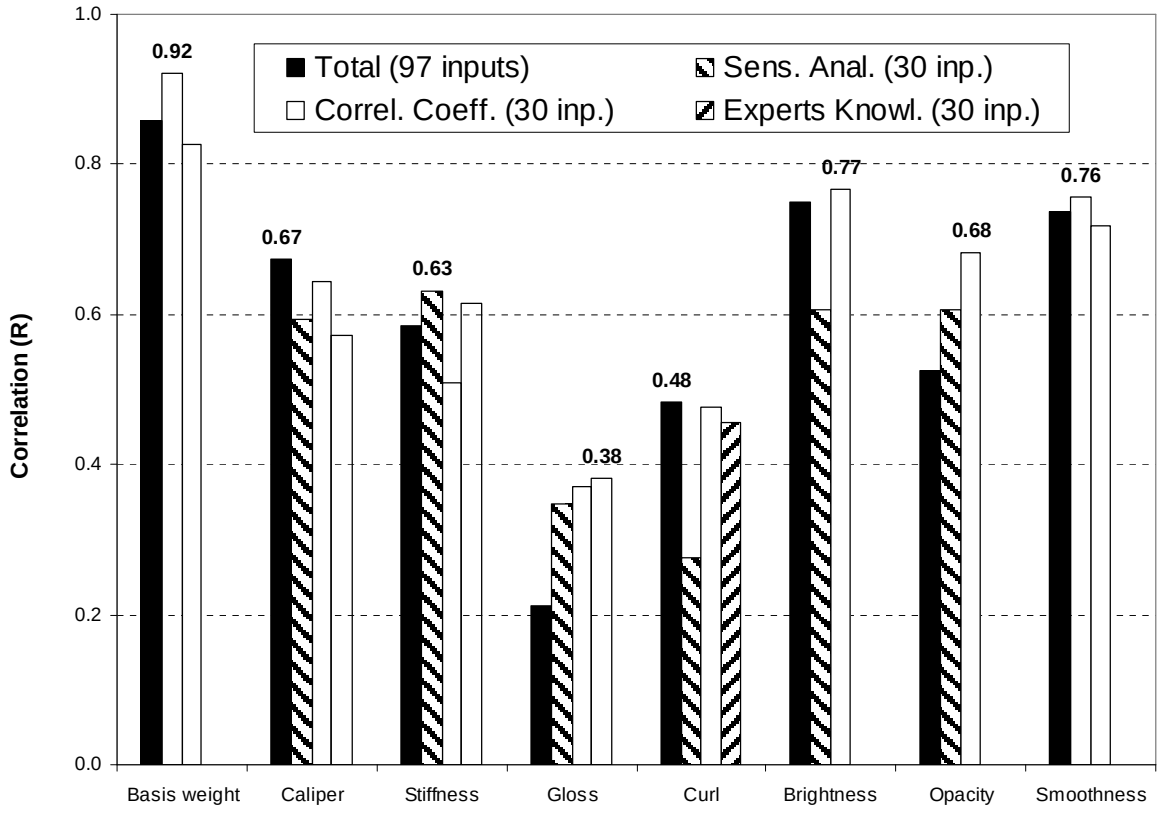
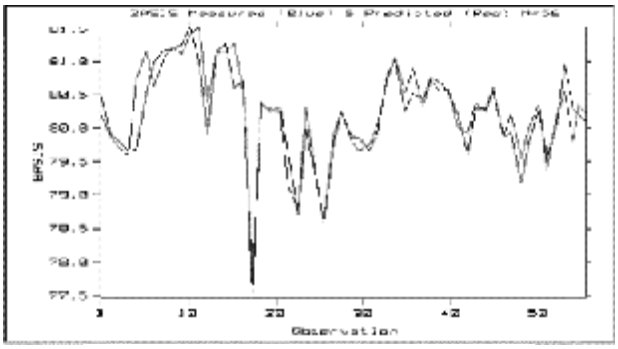


Figure 7: Comparison of models' prediction power



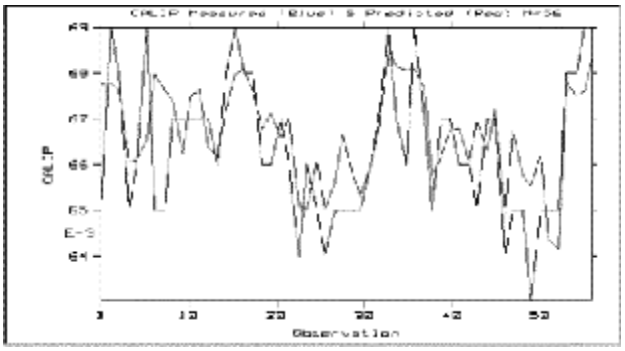
BASIS WEIGHT

RSQ: 0.850 **I/H/O: 29/18/1**

MAX: 0.52 g/m^2 **Tolerance limits:**
MIN: -1.06 g/m^2 **± 3.1 g/m^2**

Figure 8: Basis weight model performance

In Figure 8 you can see model performance for final paper basis weight. Very good matching of both curves – for measured and predicted values – and high RSQ (0.85) indicate that the network with 29 inputs and 18 hidden neurons was able to successfully learn the complicated pattern of relationships between the input variables and the final paper basis weight and also to predict fluctuation of this quality parameter to a high degree. Only with one paper sample, model prediction error (residual) exceeded 1 g/m^2. Also, tolerance limits for final paper basis weight (3.1 g/m^2) specified by the paper mill are much wider than the network model error (MAX = 0.5 , MIN = -1.1).



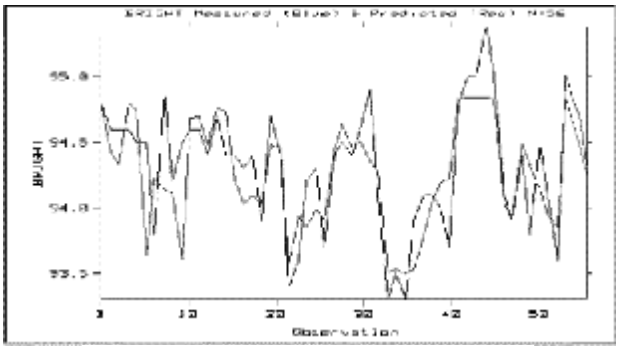
CALIPER

RSQ: 0.454 **I/H/O: 79/10/1**

MAX: 0.0024 mm **Tolerance limits:**
MIN: -0.0030 mm **± 0.0032 mm**

Figure 9: Caliper model performance

Prediction of final paper caliper (Figure 9) is, comparing to that of basis weight, significantly lower as indicated by poorer overlapping of measured and predicted values. However, although RSQ value is not so high, predictions for all of the 56 test samples still fall within the tolerance limits (+/- 3.2 microns) as specified by the producer.



BRIGHTNESS

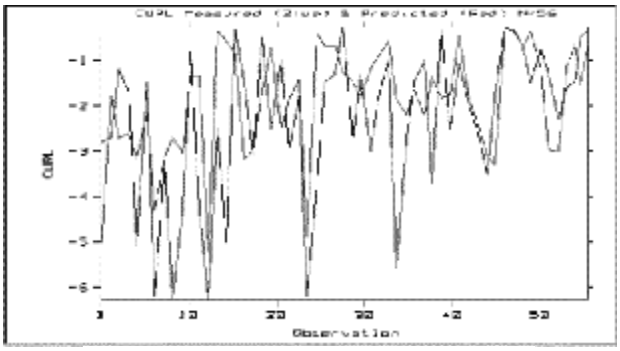
RSQ: 0.587 **I/H/O: 30/2/1**

MAX: 0.89 % ISO **Tolerance limits:**
MIN: -0.52 % ISO **± 2.0 % ISO**

Figure 10: Brightness model performance

Network configurations for prediction of paper brightness (Figure 10) consisted of 30 inputs and 2 hidden neurons. Again, all of the model predictions are better than the allowed paper mill limits (= 2.0 % ISO).

Modelling of paper curl (Figure 11) obviously proved to be a very difficult task for neural network. Matching of actual and predicted values as well as RSQ value are the poorest of all the developed models although absolute error values (+2.4 and – 4.4 mm) still fall within the allowed limits of 5.0 mm. Apparently, many things can be improved in terms of inputs selection (in this model: 97) and the appropriatness and the quality of the measuring method itself should also be examined.



CURL

RSQ: 0.234 **I/H/O: 97/8/1**

MAX: 2.44 mm **Tolerance limits:**

MIN: -4.43 mm **± 5.0 mm**

Figure 11: Curl model performance

Verification of models

Since both training and testing population were actually taken from the same database, it would be desirable to investigate how well the developed models perform when presented with samples from a new database. In order to do this, additional 160 paper tambours were obtained from paper production which took place several months later. With these tambours again all of the 107 variables were monitored. From this database 46 tambours were randomly selected and yielded a *verification* set on which the prediction power of the neural models was examined.

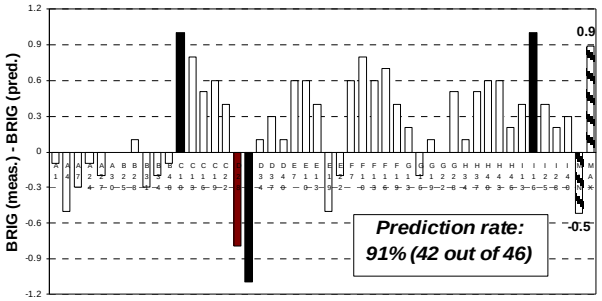


Figure 12: Verification statistics for brightness model

Let us examine how well can the already developed model for final brightness predict fluctuation of this paper property for the 46 verification samples. As seen in Figure 10, the biggest prediction errors (i.e. residuals) were +0.9 and –0.5 % ISO (last two bars in Figure 12). These values were now used as a quality criterion for evaluation of brightness prediction on 46 verification tamboures: if a residual exceeded +0.9 or –0.5 % ISO, than the brightness prediction for this particular sample was said to be poor (dark bars). Whenever residual was lower than these two numbers, the prediction was considered to be successful (white bars). Figure 12 shows that of altogether 46 verification samples only in 4 cases model was not able to predict brightness value within these limits. This means that in 91% of the verification samples brightness prediction was at least as good as in the case of test population. This is a very high number, especially if we bear in mind that the verification samples were taken from the paper production which took place several weeks later comparing to the production on basis of which the network was trained.

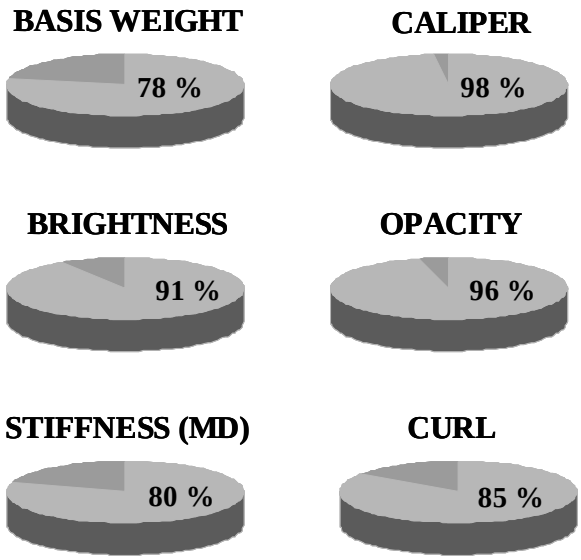


Figure 13: Prediction rates for neural models

Similar verification analysis for the rest of the models also showed that the developed networks were robust enough to successfully predict fluctuation of final paper properties. In case of final paper basis weight the rate of success – as discussed above when dealing with the verification of paper brightness – was 78% and with other quality parameters this number was even higher (Figure 13). Once again it has to be emphasized that although these models were originally trained

and tested on a different population, they proved to be very successful in predicting fluctuation of verification dataset which had not been used for training.

"What-if" scenarios

From the presented results it can be concluded that neural networks can be a powerful tool for a papermaker to investigate relationships among various paper machine settings and parameters and to immediately see their influence on final paper quality. For example, once a model for a specific end use property – let's say paper stiffness – is created, one can, by means of the simulation software, study different "what-if" scenarios: in which direction and to what extent would paper stiffness change if long to short fibre ratio, paper moisture content and size concentration all move to new values (provided that they fall within the training set range of values). Possibility to see the effect of changing technological parameters without investing a lot of production time, money and personnel in actually doing such experiments, opens new ways of thinking about product design and optimization. This approach can also be applied in the area of research, for example when planning new coating colour recipes or optimizing the existing ones.

The above described system has recently been successfully implemented in the environment of paper mill B&B Vevce.

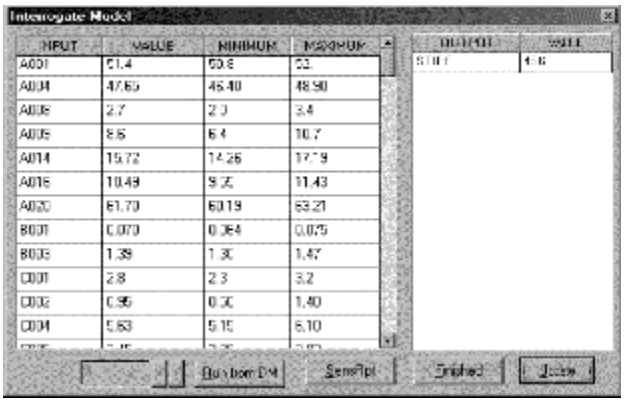


Figure 14: Screenshot from neural network simulation program

Conclusions

A feed-forward back-error-propagation neural network was trained and used for the prediction of 8 output parameters that define quality of the industrially manufactured paper based on 107 inputs. Some data details are kept confidential. Several dimensional reduction algorithms were experimented with to reduce the number of inputs to the network and improve generalization. It has been shown that the specific combination of dimension reduction strategy and network configuration for each quality parameter enables optimal prediction on the training set.

Acknowledgements

This study was conducted as a part of the doctoral dissertation at Technical University Graz and was performed in Pulp and Paper Institute in Ljubljana and in the paper mill B&B Vevce in Ljubljana. The authors are grateful to the paper mill for allowing access to their production data.

References

[1] StatSoft, Inc.; Electronic Statistics Textbook: Neural networks; Tulsa, OK: StatSoft. (2004) http://www.statsoft.com/textbook/stathome.html.

[2] Bishop CM. *Neural networks and their applications*. Rev. Sci. Instrum 1994; 65(6): 1805.

[3] *ibid*, 1819.

[4] Sarle WS., ed.: Neural Network FAQ, part 2 of 7: Learning; Periodic posting to the Usenet newsgroup comp.ai.neural-nets (2002). ftp://ftp.sas.com/pub/neural/FAQ.html

[5] Rissanen J. *Hypothesis selection and testing by the MDL principle*. Comp. J. 1999; 42(4): 260-269.

[6] Wasserman L. *Bayesian Model Selection and Model Averaging*, J. Math. Psych. 2000; 42: 92-107.

[7] Frank PM. *Introduction to System Sensitivity Theory*. Academic Press, 1978.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

A BIG LEAP FORWARD – THE NEW EUROPEAN COST ACTION E 36 "MODELLING AND SIMULATION IN THE PULP & PAPER INDUSTRY"

Johannes Kappen[1], PTS, Munich, Germany; Erik Dahlquist, MDH, Mälardalen, Sweden; Jussi Manninen, VTT, Jyväskyla, Finland; Carlos Negro, UCM, Madrid, Spain; Risto Ritala, TUT, Tampere, Finland

Abstract

The COST programme is one of the oldest EU-funded instruments to support the European scientific community. Hundreds of actions have been launched during the last decades, bringing together thousands of scientists. This year a new action concerning modelling and simulation in the pulp and paper industry has been started with a duration of 4 years. The main objective of the Action is to promote the development and application of modelling and simulation techniques in pulp and paper manufacturing processes. This is intended to eg. reduce emissions and increase the productivity and cost-efficiency of the processes. The main benefit will be a better understanding of the mechanisms of the processes and their control loops. This will help to find solutions for currently pending problems in the paper industry: improving the paper quality, optimising the wet end chemistry, enhancing the runnability and reducing emissions by improving process design, process monitoring and decision support during operation. In the long run this action should also contribute to designing superior or new product properties.

Keywords: COST, Modelling, Simulation, Pulp and Paper, Action E36

Introduction

The pulp and paper industry is currently faced with heavy economic pressure. The shut down of mills and a strong tendency to form even bigger companies clearly reflects the impact of a tight market on the paper industry. Within the mills ever less people are confronted with processes of growing complexity. Sufficient staffs of technologists are missing in many mills.

As a consequence even today trial and error appears to be the most common approach in this traditional industry. Mill personnel are fully occupied with the task of keeping the production up. No time is left for a systematic approach towards an optimised state of the production process. A lot of profit and time is lost this way.

In this context modelling and simulation will be of major importance for the paper industry in the near future. It provides the industry with new answers based on a far better understanding of the process. This know how is "built into" the process either through an optimised process design or an intelligent process control approach. In addition it can help to identify the causes for operational problems and suggest solutions.

Computational simulation in the pulp and paper industry

Problems concerned

Although the pulp and paper industry has used balancing calculations and process control for a very long time, its scope of modelling and simulation applications is not as comprehensive as that of many other modern industries. This is largely because of the complicated nature of the processes concerned in terms of raw material characteristics, the difficulty of applying real-time control tools to processes that incorporate substantial time delays, and the high degree of

[1] Corresponding author. Phone: +49 89 12146-462, Fax: +49 89 12146-36, E-mail: J.Kappen@ptspaper.de

interactions between the various production processes. As an added difficulty, some key parameters and variables of the industry's raw materials and products cannot be quantified quickly and automatically. Some of these problem fields come along with a need for an improved environmental performance of the mills. Reasons for quite many operational problems are the increasing use of recovered paper and the narrowing of water cycles.

Hope of overcoming these problems arises from developments in a wide variety of fields, including control science and process simulation (including real-time simulation). New techniques, such as multivariate statistics, software sensing algorithms and general stochastic distribution modelling and control, will enhance the controllability and permit a global optimisation of papermaking processes.

Indeed, some of these techniques have already been explored in the pulp and paper industry. Examples are data analysis tools for improved process efficiency, new formation sensors and stochastic distribution control algorithms.

Development of knowledge in pulp and paper science and technology concerning simulation

The number of publications has had a dynamic development in the past 30 years (Fig. 1). This development has accelerated during the late '90s.

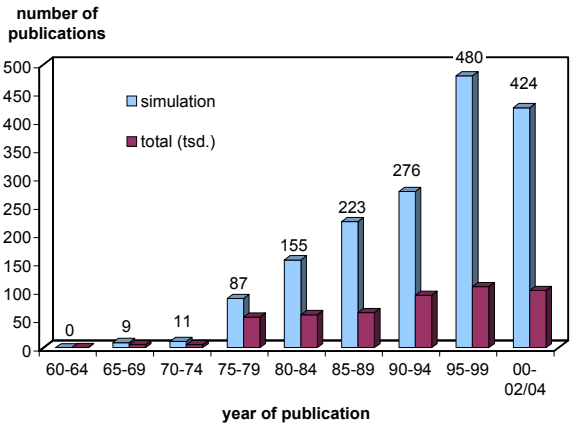


Fig. 1 Development over time of the number of publications in pulp and paper concerning simulation [1]

The most important topics handled are drying or otherwise energy related (Fig. 2). This is quite understandable since the paper industry relies far more on energy than other industries. 4.7% of the Cost is energy compared to 1.2% as an average of manufacturing [Sweden; 2]. The number of publications concerning coating, sheet formation and grade changes has had an exceptionally high growth during the past four years. This reflects most recent areas of high interest in pulp and paper.

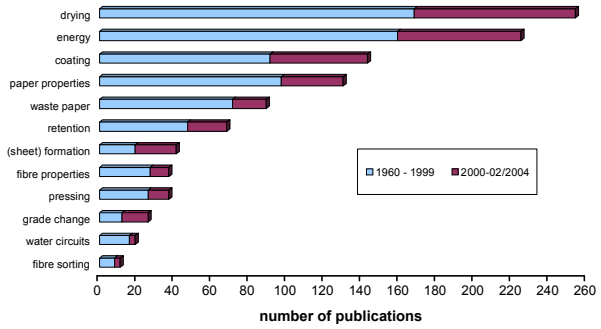


Fig. 2 Thematic distribution of publications in pulp and paper concerning simulation [1]

Evaluation of the market involved

The market for services concerning simulation activities in pulp and paper is relatively small (Tab. 1). This traditional industry is not yet using simulation to a great extent. Still, compared with approx. 2.2% rise in paper and board production between 2002 and 2003 [3] the growth rate of simulation services shows a dynamic development. This mirrors the evolvement of the scientific and technological progress described in the previous chapter.

Tab. 1 Size of market for simulation software and services in 2002 and growth rate of turnover (2002 – 2007) [4]

2002	Turnover mUS$	Growth rate %
all industries	338.7	8.5
paper industry	7.7	8.2

It can reasonably be assumed, that – as in other industry sectors [4] – software shipping has a higher share of the turnover compared to services and consultancy.

The COST funding mechanism

COST is one of the oldest funding mechanisms of the European Commission. It has been established in order to promote the exchange of scientific

knowledge within the European Community [5]. As a funding mechanism it is a predecessor of the Networks of Excellence (NoE) promoted within the 6th Framework Programme. COST is currently funded by the European Community within the Framework Programme and managed by the European Science Foundation (ESF) [6].

Objectives of the COST Action E36

The main objective of the Action is to promote the development and application of modelling and simulation techniques in pulp and paper manufacturing processes. This is intended to reduce emissions and increase the productivity and cost-efficiency of processes, for example.
The main benefit will be a better understanding of the mechanisms of the processes and their control loops. This will help to find solutions for currently pending problems in the paper industry: improving the paper quality, optimising the wet end chemistry, enhancing the runnability and reducing emissions by improved process design, process monitoring and decision support during operation. In the long run this Action should also contribute to designing superior or new product properties.
The Action has been started in January 2004 and will last until January 2008. The Action currently has 12 participating countries: Austria, Belgium, Finland, France, Germany, Netherlands, Norway, Slovakia, Slovenia, Spain, Sweden, and United Kingdom. 17 scientists represent their countries in the Management Committee. Within this group 9 professors from 7 universities are present. 8 national research institutes are participating. The work is organised in three working groups consisting of 41 scientists and technologists including 5 participants from industry.

Activites of the Action

As one of the activities to achieve the Actions scientific goals, the partners intend to compile all research projects planned, currently under way or recently finished in order to get a clear picture of the current work focus of research activities. This could also lend itself as a basis for new research and/or project related partnerships on a European level. The scientific programme will also include other activities, aimed at exchanging and disseminating ideas, findings and new knowledge as described in the following.

Work group discussion meetings

The constitutory meetings of the working groups were conducted in March 2004. Next meetings are planned to happen in September. Currently the main focus of activities is to define the priorities for the first half of the action. A more detailed description of the results reached so far is given in the following chapter concerning working group activities.

Organisation of workshops, seminars, conferences and publication of proceedings

A first conference was organised in Munich [7]. Being situated in the beginning of the COST Action E 36, it served as an ideal option to define the state, modelling and simulation in pulp and paper has currently reached. Against the background of what has been presented during the two days the following clusters could be identified:

- spreadsheet based object oriented process models,
- tools for monitoring and evaluation of online data,
- model based process optimisation and
- in smaller number, some approaches to model single process steps as the modelisation of the press- or drying section.

One large trend is to upgrade static simulation tools by integration of dynamic abilities into the simulators used. A big issue still to be solved is to define quality parameters for the validity of both, steady state and dynamic simulation models. In many papers more accurate models of the processes were identified as still missing. Thus, one important task will be to develop proper generic models of key processes that ideally will be available for common use within the industry.

Publication of targeted and periodical reports and the final report

It is planned to publish a survey on the current use of simulation software within 2004. Furthermore the publication of reports on the exchange of know-how contained in models and recommendations on suitable software tools and requirements for further software development is intended. An Action specific web page has already been set up in July 2004 [8]. Information concerning participating partners can be found there.

Working group activities

Working group A

Working group (WG) A will cover all activities concerning the use of modelling and simulation as an R&D tool, for an optimal process design, for use in operator training and trouble shooting.
To achieve this aim, the WG will cover all topics related to the modelling and simulation of the whole pulping and paper production process. This includes chemical reactions in digesters, the complex pulp washing process, the modification of fibre properties in the stock preparation process, the modelling of the complex wet end chemistry and of water loops and energy balances. Special attention will be given to the use of dynamic process simulation, real-time simulation tools and model validation tools.
The thematic focus of the working group is:

- standard model interfaces for model portability
- evaluation, calibration and validation of models
- property modelling
- water system chemistry modelling
- optimisation and simulation
- resource saving
- intelligent models
- grade-change optimisation
- data reconciliation
- diagnostics systems
- state-of-the art and who is who in modelling and simulation

The next meeting will focus on model evaluation, calibration and validation.

Working group B

WG B will concentrate on the use of simulation models during the operation phase. It focuses on model-based monitoring, simulation-based operations decision support and model-based control.
For this purpose, it is intended to look deeply into multivariable process control, fast data acquisition, high-dimensional data analysis and reduction, non-linear system modelling and multivariable system optimisation.
The thematic focus of the working group is:

- understanding the operator behaviour, decision making
- dynamic optimisation, MPDS, algorithms and applications
- combining physical simulators, grey box models and black box modelling
- model based and/or simulation based diagnostics

Next activities are intended to refine the thematic focus. The benefit and infrastructure analysis on application topics is to be explored.

Working group C

WG C intends to bring together software developers and (possible) users in order to reach agreements on the contents, features, relevance and performance of software products. Existing software packages should be evaluated. User needs and the goals of further developments are to be established. Discussions will include the development and use of software packages. In addition WG C will take care of integration aspects (simulation software in mill environments).
WG C is specially intended to take care of the knowledge exchange between the WG's and to foster the development of better simulation tools with a high compatibility across the platforms used. This WG will therefore have a different schedule, organising dedicated workshops to meet with suppliers and developers of software in order to exchange knowledge and enhance software development.
WG C has already started a survey of all partners in the COST Action E36 to gather information on current software use. In addition, all available software evaluations performed by the partners have been collected and will be processed by WG C. The results of this survey on software use will be published as a mini-booklet within 2004. Currently 13 organisations in 8 countries are taking part.
Workshops are planned to cover the topics of software evaluation, mathematical tools for data analysis, neural networks, multivariate analysis, data handling and pre-processing before simulation and on requirement specifications for future simulation tools.

Summary and Outlook

The COST Action E36 is a promising approach to foster the development of modelling and simulation in the pulp and paper industry. The expertise of the European paper industry is gathered in this action. It will contribute heavily to

the acceptance of modelling and simulation approaches in the pulp and paper industry.
During the next years all possible users of modelling and simulation technologies in the pulp and paper industry will have to define their position in terms of staff, software and total involvement.
The most important task for all software and solution developers will be to show the possible economical benefit the pulp and paper industry has by using the tools developed.
A network of excellence has been created and will possibly lead to numerous successful follow up activities as has been proven by other COST Actions.

[1] Combined search results in literature databases Papertech (www.ptspaper.de) and Pirabase (www.pira.co.uk)

[2] Bergman MA, Johansson P. *Large investments in the pulp and paper industry: a count data regression analysis.* Forest Economics 2002;8(1): 29-52.

[3] *Production Statistics 4th Quarter.* Brussels: CEPI, 2004

[4] *Process Simulation and Optimization.* Dedham: ARC, 2003
With kind permission of: ARC Advisory Group (www.ARCweb.com)

[5] COST homepage: http://cost.cordis.lu

[6] CORDIS homepage: www.esf.org/

[7] Kappen J., Bienert Ch. *Simulation and Process Control for the Paper Industry.* Munich: Proceedings PTS MS 441, 2004.

[8] www.costE36.org

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

EXPERIENCES AND CHALLENGES IN DYNAMIC SIMULATION OF PAPER PRODUCTION PROCESSES

Jari Lappalainen*, Petteri Kangas, Jouni Savolainen and Kaj Juslin
VTT Technical Research Centre of Finland
Industrial Systems
Espoo, Finland

Jean-Peter Ylén
Helsinki University of Technology
Control Engineering Laboratory
Espoo, Finland

Juha Pursimo
Tampere University of Technology
Automation and Control Institute
Tampere, Finland

Abstract

Paper production processes have plenty of special features and challenges from the modelling and simulation point of view. Questions arise on fibre suspension characterisation, flow behaviour, mixing, cleaning and fractionating, retention, drainage, drying, wet end chemistry, paper quality properties, sheet thickness and cross direction, to mention but a few. Each of these questions require different kind of attention, thus challenging the model developer, as well as a user of simulation tools, to find the best workable solutions and practices for the needs in hand. Many of the issues above are common for all simulation users, whether you are building a steady state or a dynamic model, or a computational fluid dynamics (CFD) simulation. This paper discusses some of the above mentioned challenges focusing on modelling of paper and board making and real-time (or faster) full-scale dynamic simulation. Illustrative simulation examples are given.

Introduction

Use of simulation in paper industry is on the rise. Simulation is used, for instance, in process and control system design evaluation and optimisation, control system testing, process analysis and troubleshooting, as well as in operator training and support.

In papermaking, the raw material is very heterogeneous. It is transported both as hydraulic flow and as a paper web. One-dimensional simulation of approach system covering sheet machine direction (MD) variables after the head box is enough for most of dynamic simulation needs. In some cases there is a need to simulate flow systems in three dimensions, or capture the thickness and cross direction of the paper sheet as well. These are examples of special requirements for papermaking simulation.

Wasik [1] analysed simulation from pulp and paper perspective, giving requirements for good simulation software in the field, and examples of applications. This paper continues on the same road, discussing aspects and challenges, giving examples of modelling and simulation in the paper

* Corresponding author: Phone: +358 9 456 6428, Fax: +358 9 456 6752, E-mail: jari.lappalainen@vtt.fi

industry. The experiences are based on the development and use of the Apros Paper simulation platform (formerly known as APMS [2, 3]).

Raw materials

The main papermaking raw material, fibres retain very non-uniform characteristics. Fibres are transported and processed together with water, typically in consistency range from 0.5 to 4 %. A significant amount of air is present in many parts of the process, though this is usually neglected in simulations. Fibres have a wide distribution of different lengths and shapes, greatly depending on the wood species, and the way of pulping, e.g., chemical or thermo-mechanical pulping or re-cycling. The following two approaches have been used to describe the fibrous components carried in simulated flows:

- Fibres with similar pulping history (e.g. TMP) form a component
- Fibres with specific dimensions (e.g. fibre lengths between 1.0 and 2.0 mm) form a component.

Figure 1 illustrates these two methods. The first method makes it easy to define the boundary values needed for the simulation, thus fibres from each source are defined as a new component and the mass fraction is obtained from standard consistency measurements. Broke is naturally a mixture of these "new fibres". The second method is based on the physical properties of fibres and gives better basement to develop detailed, mechanistic based unit operation models. But, on the other hand, it is not straightforward to define the composition of the model boundaries, nor get data for validating the simulation results. An example of this approach was presented by Yli-Fossi *et al.* [4]. They divided the stock into six components (water, fines, short fibres, long fibres, and two filler types) and searched the model parameters based on large measured data. It is more common to simplify the method 2 further and divide the stock to water, fibre, fines and filler. Furthermore, in many reported simulation cases, the suspension is simplified to consist of water, fibre and filler.

Besides fibres and fillers there may be a number of other components fed into the suspension, such as retention aids, starch, foaming agents, biocides or dyes. These have no practical influence on the mass balance, but they may have a remarkable effect on retention, dewatering, foaming, biological activity, quality variables, etc. It depends on the simulation accuracy and needs whether these are carried in the flow network as independent components. Many times lack of quantitative information on their effects makes it useless to incorporate them in the simulation.

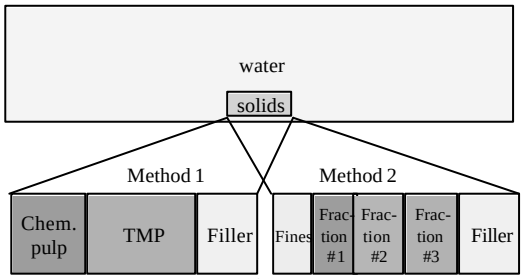


Figure 1: Illustration of two typical approaches used to define the solids in stock suspension.

Flow and mixing

Besides describing the components in fibre suspension for simulation, a relevant question is how to calculate flows in pipes, valves, etc. The rheology of stock suspension is greatly different to water. Besides the velocity and pipe diameter, the pressure drop over a certain length of a pipe depends on, e.g., consistency, pulp type, temperature, freeness, pipe roughness, fibre length/thickness ratio, filler content and air content. Figure 2 shows a typical example of a stock velocity - pressure drop curve. The dip in the friction loss curve is an especially interesting feature of fibre suspensions. In plant design, correlations and design rules (e.g. [5] have been used to determine piping dimensions. No easy, generally accepted, unambiguous way to calculate pressure drop - velocity behaviour for different pulp types exists so far. However, in most simulation studies in paper making, it is adequate to calculate the stock flow using pure water properties. In addition, the question of accurate friction losses is often irrelevant in paper making

simulation, thus most of the simulators do not provide a rigorous pressure-flow network solver, but use a modular sequential flow solver. Some simulation tasks require pressure-flow solution, for other tasks it only slows down unnecessarily the model build-up and simulation. Thus, the simulation platform should offer accuracy locally where it is needed.

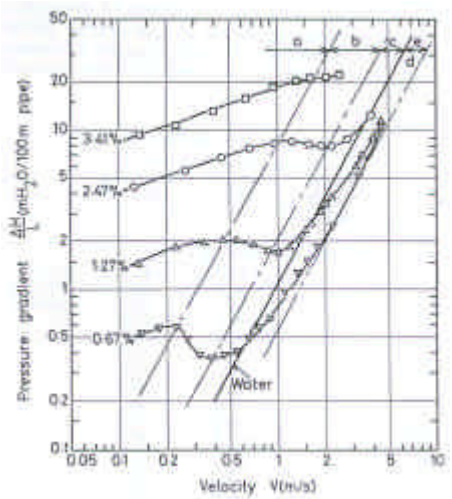


Figure 2: Pressure drop curves for unbleached sulphate pulp (4 consistency levels) and for water [6].

Another flow related issue is how the delay and mixing of stock in pipes should be handled. The common approach is to use ideal mixing in tanks and plug flow in pipes. In most cases good results are obtained by combining these two basic types, as illustrated in Figure 3. The picture on the left side is measured in a 3-ply paper board process [7]. The tracer was fed into the thick stock lines just after the machine chests, and measured in head boxes. The upper picture presents the corresponding simulation with Apros Paper. The pipe lengths were estimated at the mill and the delay calculation is based on plug flow with a small percentage of mixing involved. The wire pits are modelled with two ideal mixers in series. In this simulation the flows are solved with simultaneous pressure-flow network solver.

The paper production process has plenty of dilution steps, i.e. the stock consistency is controlled to a lower value with low consistency process water. Different fibrous raw materials, mineral based fillers and chemical additives are proportioned and mixed carefully to form the best furnish for the desired product.

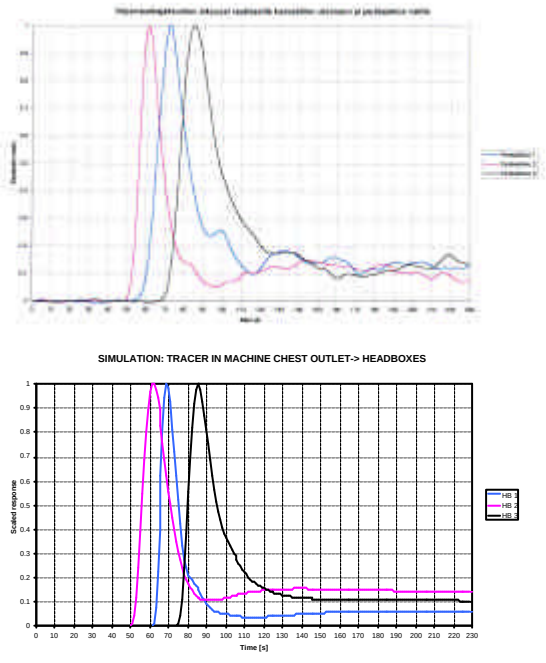


Figure 3: Measured [7] and simulated tracer experiment in a 3-ply board machine.

In practice, the degree of mixing is often insufficient, leading to disturbances in paper quality. Lack of fundamental knowledge prevents simulation developers to incorporate models for different types of non-ideal mixing.

Screening

Screens and centrifugal cleaners are used to remove unwanted parts, e.g. shives, from the raw material in short circulation. Particles in the feed flow are divided into two or more streams based on differences in physical properties such as fibre length, width and coarseness. In addition to removing bad particles out of the main stream, they also perform fractionation. In short circulation we see this as reject thickening. The big opportunity, however, is to improve the raw material usage with fractionation, by directing each fraction to the purpose it is best for. To be able to optimize the fractionation, one must understand the underlying mechanisms thoroughly. The last ten

years many research groups have studied screening in laboratory or pilot scale producing valuable information for modelling. The wide distribution of physical properties of fibres and the numerous operational parameters of the equipment and process make the modelling task very challenging. A recent review on the progress in the field can be found in [8]. Today, screening is not an operation that can be predicted accurately at a mill. The basic challenge for using the models developed is that they are equipment specific and stock specific. At mill sites, the measurements for validation are missing too. The situation is not much better, even if we were satisfied with modelling reject thickening only. Namely, the need to include fibre length distribution in simulation (being laborious to validate in full scale simulation) seems to be essential for realistic thickening calculation [9]. This is no surprise, because thickening is a consequence of fractionation. Accordingly, model builders often end up with a solution of fixed splitting coefficients. In most short circulation simulation studies fractionation is not a key point, so a simple screen or a centrifugal cleaner model does not ruin results, for instance the dynamics of head box consistency.

Forming

Head box spreads the stock flow from the approach pipe to the whole width of the machine: the fibre mat is formed. The material flow is abruptly changed from a hydraulic flow to continuous sheet that must be controlled to have uniform properties at the reel. Besides the "trivial" requirement to cover the sheet machine direction (MD) in simulation, decisions whether, and how to simulate the sheet thickness and cross direction (CD) as well, become relevant.

The forming section is a key operation in paper making. However, attempts to describe behaviour of the fibre mat on the wire mathematically have not been very successful, at least they are not utilized in full-scope dynamic simulators. The issues such as the stochastic nature of the raw material and fast moving wire or twin-wire make the hydrodynamics at the forming section very complex. The important phenomena, retention and drainage, are affected by wet end chemistry as well. The detailed models developed are very much machine, even paper grade specific. Also condition of the equipment (e.g. wire fabrics) is significant. It is difficult to apply even simple models in a production machine, because there are hardly any on-line measurements available in the forming section. For these reasons, fixed retention coefficients and water removal are widely used in simulation studies.

Sheet thickness and cross direction

Detailed modelling and simulation of paper sheet in three dimensions throughout the whole paper machine, is a great challenge. Needs to cover all three dimensions exist. Sheet thickness is important when one aims at detailed dewatering models in wire, press and drying parts. Quality variables are measured and controlled in paper cross direction (CD), thus an extensive simulator should offer all the manipulated variables in CD and produce realistic responses to be measured by the simulated scanner.

Modelling of paper sheet thickness direction phenomena has been most successful in wet pressing and drying area. Many of such models can be found in literature, see e.g. reports by Kataja *et al.* [10] in wet pressing and Sidwall *et al.* [11] in paper drying. These models solve mass and heat transportation in, out and inside the sheet. Usually such detailed models are stationary, and the scope of the simulation is limited to single unit operation. Dynamic, full scope simulators require lighter solutions in unit operation modelling. The reasons are very practical such as limited time to configure a single piece of equipment and the general requirement for real-time or faster simulation speed. For these reasons Apros Paper uses simple wet pressing models (like Decreasing permeability model [12]) and single point sheet model (homogeneous sheet in thickness direction) in the drying part simulation. However, Apros Paper calculation has been developed to take into account the uneven temperature and moisture. Example of

a drying section profile, simulated and measured, is presented in Figure 4.

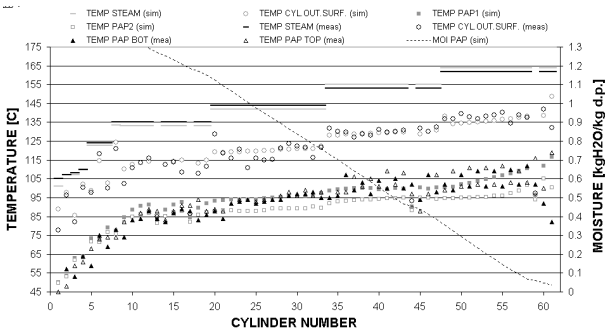


Figure 4: Measured and simulated profiles at a drying section of a board machine.

Only a few reports on modelling and simulation of machine CD have been presented and the simulation scope has been very limited. Probably the only full scale application has been presented by Nissinen *et al.* [13] who reported modelling and simulation of web forming process with dilution head box. The simulator provides a single tool to study paper machine MD and CD variables simultaneously. A grade change example is presented in the paper. This simulator (Metso's in-house extension in the Apros Paper simulation environment) has been used and further developed actively.

In the simulator presented in [13] the full sheet CD model was limited in the forming section. Only the sample that was calculated to be seen by the traversing measurement head was stored and delayed. At that time Nissinen *et al.* wrote "it is evident that the full profile information cannot be held in the simulator's memory resident database." Today the simulator does exactly this, provides the CD information of the entire sheet width on the whole MD length, and still runs faster than real-time. This approach gives a great framework to build an extensive paper machine simulator with detailed sub-models of forming, wet pressing, drying, finishing, etc. both in MD and CD. The simulator enables studies on operations such as machine speed changes or changes in the speed differences between drive groups. The simulator is linked with the metsoDNA control system and used in testing novel control strategies. Figure 5 shows a sample experiment with the simulator. Dilution valve opening profile and a corresponding scanner measurement profile of basis weight are presented. After the situation in the left side picture, some of the dilution valve positions were manually altered. After some tens of seconds of simulation (the MD delay from the head box to the scanner), the right hand pictures were captured. The arbitrary changes made in the valve positions and the corresponding effects in basis weight profile can be clearly seen.

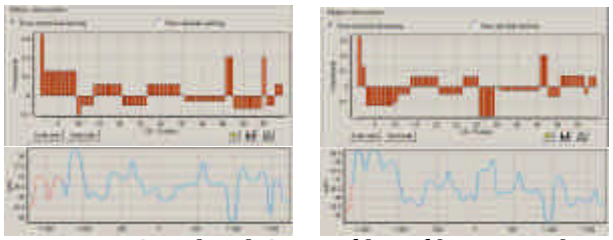


Figure 5: Simulated CD profiles of basis weight.

Chemical state

Traditionally in dynamic simulation of paper making, chemical phenomena have not been included. Research in the field is active, and this limitation is gradually being removed. However, to create a general model of chemical state in paper process is practically impossible. Special models have to be made for different purposes, e.g. Koukkari *et al.* [14] have modelled chemistry of a neutral paper machine short circulation. This model is valid under certain constraints (when certain components are involved in the wet end system).

Many calculations of chemistry are done as steady state calculation: kinetics is neglected. In real paper processes the reactions does not always reach the balance. One option is to calculate reactions in single phase as steady state reactions and add kinetics between the phases. Ylén [15] employed this approach when estimating the pH in short circulation slurries.

A general problem in modelling chemical state of a paper making process is that the related on-line measurements can only capture a faint projection

of the entire chemical state of the system. Measurements – such as pH and conductivity – depend on all the chemical and physical components in the system and there are an infinite number of component combinations, which produce the same measured value. Once this fact is recognized, it becomes obvious that the process can not be operated optimally based on only the available measured values. For example, the same measured situation in plant – say pH – reacts very differently to chemical additions on different occasions even though initially all measured values are the same.

There are significant problems with off-line measurements as well. The chemical and physical phenomena never reach equilibrium in practical situations due to the different time constants varying from nanoseconds to several days and even years. When a sample is taken from the plant, it will continue its pursue of equilibrium and when the analysis is finally performed the measured off-line values do not correspond to the real situation in the plant. Furthermore, the sample changes due to external conditions (temperatures, pressures, interaction with surrounding air, etc) and if the situation at the plant changes rapidly the sample taken represents only one situation at one particular time instant.

These issues make realistic simulations of chemical state a challenge. In fact, the same problem (that on-line measurements offer only a limited projection of the actual state of the process) apply to many basic measurements in pulp and paper processes, for example freeness, kappa number and even consistency. If modelling is based on this kind of limited information, accuracy of the simulation (e.g. in on-line prediction) can as well vary quite remarkably depending on the operational state. Probably the best understanding of the state of the system (in addition to developing better on-line measurements) could be reached by combining all available on-line and off-line measurements to modelling and simulation structures in a Kalman filter type approach. This estimation/measurement system could evaluate the states which could not be measured directly.

Quality

Paper quality variables can be divided in two different categories:

A. Quality variables, which are based on clear physical properties of paper, e.g. grammage, moisture or density.
B. Quality variables, which can not be defined using physical properties, e.g. freeness or formation.

Quality variables in the category A can be defined using first principle models. These variables are unambiguous. For example the study by Lappalainen *et al.* [16] showed accurate results for grammage and moisture of paper.

The variables in the category B are, however, much more difficult to estimate. It is known that there are an infinite number of component combinations, which produces the same Freeness number. Many of the quality measurements used in paper industry are not unambiguous. A lot of sophisticated measurement methods are developed e.g. for tensile strength. Strength of fibres and bonds can be measured. However, the quality measurement used in industry is still tensile index.

Statistical models can be developed to predict paper quality variables, though in many cases only in qualitative level. To use such a model, one must involve the effects of process delays in the prediction. Two approaches have been presented:

1. Quality parameters of stock and paper are modelled and calculated after every unit operation. Jones and Nguyen [17] have used this methodology.
2. Quality parameters are not modified after every unit operation. Instead, the operation /control variables of the unit operation (e.g. refining energy) are placed into the stream. The paper quality variables are calculated later using these variables and the statistical models developed based on process measurements. Kangas [18] proposed this idea of modelling paper quality using fibre-processing history.

Figure 6 presents a simulation in which refining energy is elevated. Refining energy is put into

stream and transported with material. The elevated refining energy comes first to the machine chest. Later, the same effect can be seen in the machine screen, in the head box and finally at the reel.

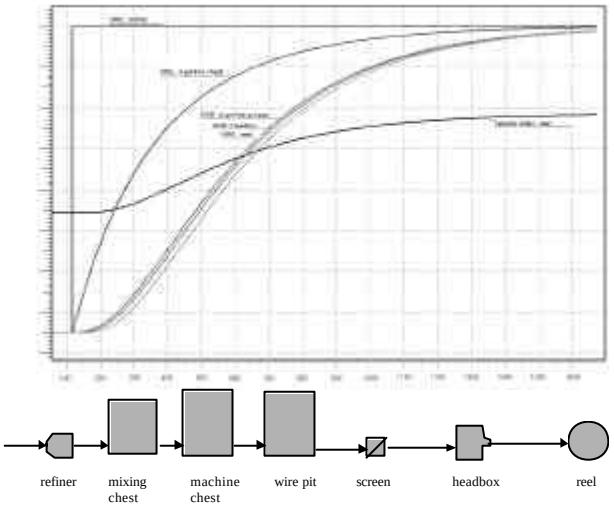


Figure 6: Fibre processing history is transported with stream and used at reel to calculate a paper quality variable.

In this experiment, the value of refining energy (as an example of an item in fibre-processing history) is transported through the whole process. At reel there is a statistic model which calculates the effect of elevated refining energy to tensile strength of paper. This model can include other parameters affecting the tensile strength as well: fibre orientation, draws, chemical dosage, etc.. Using this novel idea of fibre-processing history the quality properties of paper are estimated when needed, not everywhere in process network. The approach suits well also for statistical quality models which are based on combining sampled quality measurements with process measurements from mill control/information system.

Discussion and conclusions

Computational simulation of paper making, both static and dynamic, has been studied and applied since the 70's, though papermakers have not been in the leading group to utilize simulation in industry. There is no external force (safety, legislation, etc.) for using simulation as compared to for instance nuclear industry. Secondly, paper making is full of processes where first principles models and accurate predictions are still far away in the future. Pure water in power processes is much easier to simulate than fibre suspension. Better on-line measurements would help here. The key difficulty is that the raw material is very non-uniform. The on-line measurements can offer only a limited projection of the actual state of the process. It is extremely difficult to get general knowledge for needs of dynamic models from a paper mill. In the future, CFD simulation may give a boost to model development of full-scope dynamic simulation. Today, CFD simulation with detailed 3D models of papermaking equipment has limitations on e.g. fibre characterization and it still needs a lot of computing power. However, the idea of making reliable experiments with a CFD model, for instance on fractionation in pressure screens or centrifugal cleaners, is very attractive. Recently, many interesting reports in the CFD simulation arena have been published; see e.g. [19] and [20].

Despite of all the difficulties mentioned in this paper, the interest and use of simulation is increasing in paper industry more rapidly than the industrial average [21]. One reason is the unquestionable benefit that dynamic simulation offers, even if we could not predict exact behaviour of every piece of equipment, namely the possibility to see and study the papermaking process as a whole [22]. Process and controls form together a complex integrated system with many interactions and re-circulations. The time span of the process phenomena varies from milliseconds to hours. Recently, the papermaking process is undergoing a significant change. Machines are getting either very large or compact and agile. Production and quality margins are getting smaller and smaller. Modern mills search continuously for improvements through advanced process control. Automation takes a more important role. Consequently, it becomes more difficult for new operators to understand the production system deeply enough, for example to act correctly in new, exceptional situation. Training and supporting simulators can provide help here. In this scope, it really seems that a simulator meeting the various demands - one by one - is needed.

Contradictory to the proposed trend of increasing simulation activity is the fact that engineering staff at mill sites has been reduced. It is hard to find time and suitable people to get familiar with simulation tools. The diffusion of simulation technology to the mills takes mostly place via engineering universities, and equipment and automation suppliers. Dynamic simulation in education should be increased, not only because it is one of the key technologies in the future, but because by using simulation examples and exercises the often theoretical material of process and control technology is made more intuitive and interesting for students.

Acknowledgements

We want to acknowledge Metso Automation for the permission to demonstrate the CD simulator and Stora Enso for the permission to use the figure with the tracer measurement.

References

[1] Wasik, L.S. *Simulation, a pulp and paper perspective.* In: Proceedings of the 43th SIMS Simulation and Modelling Conference, Oulu 2002, pp. 11–20.

[2] Laukkanen, I., Juslin, K., Tuuri, S., Lappalainen, J., Niemenmaa, A., *Object-Oriented Modeling and Dynamic Simulation of Paper Machines.* In: Proceedings of the 37th SIMS Simulation Conference: Simulation in Theory and Practice, Lyngby 1995, pp. 46-53.

[3] Niemenmaa, A., Lappalainen, J., Laukkanen, I., Tuuri, S., Juslin, K., *A multi-purpose tool for dynamic simulation of paper mills*, Simulation Practice and Theory 1998;6:297-304.

[4] Yli-Fossi, T., Kosonen, M., Toivonen, M., Karaila, I., Lautala, P., Huhtelin, T. *Modelling of stock fractions for a dynamic paper machine simulator*, Proceedings of the Scandinavian Paper Symposium, Helsinki 2002, 6 p.

[5] TAPPI. *Generalized method for determining the pipe friction pressure loss of flowing pulp suspensions*, TAPPI Technical Information Sheets, TIS 408-4, TAPPI Press, Atlanta 1981, 5 p.

[6] Moller, K., Norman, B. *Calming lengths in pulp suspension flow*. Svensk papperstidning 1975;78(16):582–587.

[7] IndMeas. *Internal measurement report*. Stora Enso, 1997.

[8] Gooding, R., Olson, J., Roberts, N. *Parameters for assessing fibre fractionation and their application to screen rotor effects*. TAPPI JOURNAL 2004 3(3) 20 p.

[9] Julien Saint Amand, F., Perrin, B. *Screening: experimental approach and modelling*. In: Proceedings of the 1998 TAPPI Pulping Conference, TAPPI Press, Montreal 1998, pp. 1019–1031.

[10] Kataja, M., Hiltunen, K., Timonen, J. *Flow of water and air in a compressible porous medium: model of wet-pressing of paper*. J.Phys.D:Appl. Phys. 1992;25(7):1053-1063.

[11] Sidwall, S., Bond, J.-F., Douglas, W.J.M. *Industrial validation of a multiple technique paper drying simulator*. In: Proceedings of TAPPI 1999 Engineering/Process & Product Quality Conference, TAPPI Press, Anaheim 1999b, pp. 271–301.

[12] McDonald, D., Amini, J. *Predicting the press dewatering of heavyweight grades*. TAPPI JOURNAL 2000;83(2):79-82.

[13] Nissinen, A., Tuuri, S., Rinta-Kokko, P., Airikka, P., Huhtelin, T., Lautala, P. *Modelling and simulation of web forming process with dilution head box*. In: Proceedings of the 19th IASTED International

Conference Modelling, Identification and Control, IASTED, Innsbruck 2000, 5 p.

[14] Koukkari, P., Pajarre, R., Pakarinen, H., Salminen, J. *Practical multiphase models for aqueous process solutions*. Industrial and Engineering Chemistry Research 2001;40(22):5014-5020.

[15] Ylén, J.-P. *Measuring, modelling and controlling the pH value and the dynamic chemical state*. Doctoral thesis, Technical University of Helsinki, Control engineering laboratory, Espoo 2001, 159 p.

[16] Lappalainen, J., Myller, T., Vehviläinen, O., Tuuri, S., Juslin, K. *Enhancing grade changes using dynamic simulation*. TAPPI JOURNAL 2003 Online Exclusive; 2(12):13 p.

[17] Jones, G., Nguyen, X. *Analysis and simulation of property development in forming newsprint.* TAPPI JOURNAL 1990; 73(7):160-168.

[18] Kangas, P. *Modelling effects of beating on paper properties*, Master thesis, Technical University of Helsinki, Forest Industry Department, Espoo 2001, 114 p.

[19] Hammarström, D., Hämäläinen J.P., Dahlkild, A. *CFD simulation of fibre suspensions*. In: Proceedings of VIII Suomen Mekaniikkapäivät, Espoo 2003, pp. 71–76.

[20] Statie, E., Salcudean, M., Gartshore, I., Bibeau, E. *A computational study of particle separation in hydrocyclones*. Journal of pulp and paper science 2002; 28: 84–92.

[21] Kappen, J. *Simulation in the paper industry - an overview*, PTS News 2002; (9): 18-22.

[22] Kokko, T. *Development of papermaking process based on integrated process and control design.* Doctoral thesis, Technical University of Tampere, Automation and Control Institute, Tampere 2002, 141 p.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

INTEGRATED MODELLING OF THE WELDING, MACHINING AND HEAT TREATMENT PROCESSES

Jesper Thorborg* **and Jesper Hattel**
Technical University of Denmark
Department of Manufacturing Engineering - Process Modelling Group
2800 Lyngby
Denmark

John Hald
ELSAM/Energy E2, Department of Manufacturing Engineering
Technical University of Denmark

Abstract

The sequence of the primary welding process and the following secondary processes machining and heat treatment has been modelled to predict the residual conditions in a steam turbine valve. The different process steps are coupled in order to transfer the residual conditions from preceding processes to the ones following. The integrated model has been used to examine the rupture in the weld zone between P91 and stellite. An accurate prediction of the stress level in the weld zone is imperative for the assessment of the quality in the component and by that having a tool to minimize the risk for breakdown and need for expensive maintenance. Both classical time independent and time dependent plasticity models have been used to describe the different material behaviours during the different process steps. The description of the materials is highly temperature dependent. The welding process is modelled by adding material to the calculation domain while the machining process is modelled by removing material from the calculation domain to redistribute the stress fields. An in house iso-parametric FEM-code is used for the implementation of the material models and the discretization of the appropriate addition and removal of material. The return mapping algorithm is used for the time discretization of the time independent plasticity model and a Norton's power law model is used for the time dependent model. Results from the numerical calculations are presented and different combinations of process parameters and material data are examined.

Keywords: Integrated modelling, Plasticity, Creep, FEM, High pressure steam valve

Nomenclature

σ_{ij}	Stress tensor [MPa]
u_j	Displacement fields [m]
ε_{ij}	Total strain tensor [-]
ε_{ij}^{el}	Elastic strain tensor [-]
ε_{ij}^{pl}	Plastic strain tensor [-]
ε_{ij}^{cr}	Creep strain tensor [-]
ε^{th}	Thermal strain [-]
Q	Activation energy $[\frac{J}{K\ mol}]$
R	Gas constant $[\frac{J}{K\ mol}]$
T	Temperature $[^{\circ}C]$
ρ	Density $[\frac{kg}{m^3}]$
c_p	specific heat capacity $[\frac{J}{K\ kg}]$
k	Thermal conductivity $[\frac{W}{m\ K}]$
σ_Y	Yield stress [MPa]
E	Young's modulus [MPa]
ν	Poission's ratio [-]

*Corresponding author. Phone: +45 4525 4716, Fax: +45 4593 4570, E-mail: jt@ipl.dtu.dk

Introduction

A maintenance inspection of a high pressure regularizing valve in the turbine section of the Danish power plant "Skaerbaek 3" exposed a critical and total breakdown of a stellite layer welded on a P91 base material. A crack in the interface zone between the two materials was fully developed and the entire layer of coated stellite material could be removed with light tools, see figure 2. The breakdown is of course critical since the unit must be stopped during repair and the number of similar units makes it critical to get a good damage assessment to understand the nature of the breakdown. In this part of the damage assessment the focus is put on the manufacturing stages, i.e. the initial welding of stellite on the P91 material, the machining of the top layer weld-material and the subsequent heat treatment of the entire valve. To analyse the influence of these processes, an in-house FEM code has been further developed and used to model the thermomechanical history of the material. To get useful results from the numerical analysis, it is imperative to couple the results from the different process stages in order to accumulate the full load history of the material. Therefore, the residual conditions from the welding and machining process are used as initial conditions for the subsequent heat treatment analysis. The residual conditions from this calculation will be used for an in-service calculation later on.

The chain of processes and necessary coupling in the numerical analysis can be illustrated by the following figure

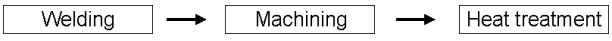


Figure 1: Process modelling stages

In the numerical model the first stage of welding is modelled by adding material to the discrete model in steps as the weld material is added in the physical system. During the machining process parts of this material are removed again from the calculation domain and the stress redistribution is calculated. Finally, the heat treatment process is modelled by increasing the temperature to a specified level and holding this temperature for a specified amount of time. During this time of elevated temperature the stress relaxation due to creep is calculated and after cooling to ambient temperature the quality of the component and effects of manufacturing stages can be evaluated.

The results are evaluated by monitoring the development of the different fields in several nodes selected in the cross section of the valve. These nodes are located in a quasi stationary part of the structure that more or less represents the overall load conditions in the weld zone.

The enmeshment of the structure is done entirely by 8 node brick elements with 8 Gauss points for the numerical integration. To speed-up calculation time only one fourth of the structure is modelled. This is of course obtained by setting appropriate boundary conditions in the symmetry planes, i.e. adiabatic boundary conditions for the thermal calculation and perpendicular constrains for the mechanical calculation. The rest of the boundary conditions are generally free for the mechanical part and modelled by a convective boundary condition in terms of an equivalent heat transfer coefficient for the thermal part.

For the weld simulation the constitutive model is based on a classical time independent J_2 incremental theory with a temperature dependent yield surface. For this part of the analysis the temperature dependency of the yield stress is considered to be the first order effect. During the heat treatment process the time effects are considered by using a power law creep model. The temperature dependency of the strain rate is described by an Arrhenius expression to model creep at elevated temperatures. The implementation of both models are based on a backward Euler time integration scheme to ensure a stable numerical algorithm with sufficient accuracy. These implementations require Newton Raphson equilibrium iterations to preserve global force equilibrium.

Theory

The numerical analysis is based on the finite element formulation to solve the thermo-mechanical problem. The two physical systems of heat balance and force equilibrium are described by the governing heat transfer equation and the equilibrium equations for the thermal and the mechanical analysis respectively. Both systems of equations are shortly presented. The considered constitutive models and the numerical implementation of these are presented, i.e. rate dependent and rate independent plasticity and the time integration of these.

Heat conduction is assumed governed by the Fourier

law, Carslaw and Jaeger [1]. Together with the source term from the process, the (transient) governing equation for temperatures becomes

$$\rho c_p \dot{T} = (kT_{,j})_{,j} + \dot{Q}_V \tag{1}$$

where T is the temperature; ρ is the density of the material; c_p is the specific heat; k is the heat conductivity; and $\dot{Q}_V$ is the is the heat supplied externally into the body per unit volume from the welding process.

The mechanical model is based on the solution of the three governing partial differential equations of force equilibrium, Olhoff [2], Hattel [3] and [4]. In tensor notation, these are written as

$$\sigma_{ij,j} + p_j = 0 \tag{2}$$

where p_j is the body force at any point within the volume and σ_{ij} is the stress tensor.

Constitutive models

The material behavior during the welding and heat treatment processes is indeed governed by the thermal load and the temperature influence on material data and constitutive relations. For the higher temperature regions the considered materials generally behave non-linearly and plasticity and creep effects must be considered to get an appropriate description of the material response. Still the fundamental assumption of Hooke's law forms the basis of the constitutive relation by including non-linear strain contributions in the decomposition of the strain expression.

The fundamental Hooke's law and the decomposition of the strain tensor are given as, e.g. Weiner and Boley [1]

$$\sigma_{ij} = L_{ijkl}\varepsilon^{el}_{kl} \quad \textit{and} \quad \varepsilon_{kl} = \varepsilon^{el}_{kl} + \delta_{kl}\varepsilon^{th} + \varepsilon^{pl}_{kl} + \varepsilon^{cr}_{kl} \tag{3}$$

The thermal strain is in nature part of the elastic system and gives a contribution to the dilatatoric part of the system which forms the compressible behavior of the material, i.e. pressure and density changes. For the implementation of non-linear material models it is convenient to split the stress and strain tensors in the deviatoric part, s_{ij} and e_{ij}, and the dilatatoric part σ and ε, e.g. [5], Tvergaard [6] and Lubliner [7] . Hence, the non-linear models are implemented in the deviatoric system only, to fulfill the requirement of incompressible behavior.

$$\begin{aligned} \sigma &= \tfrac{1}{3}\sigma_{ii} \\ \varepsilon &= \tfrac{1}{3}\varepsilon_{ii} \end{aligned} \qquad \textit{and} \qquad \begin{aligned} s_{ij} &= \sigma_{ij} - \delta_{ij}\sigma \\ e_{ij} &= \varepsilon_{ij} - \delta_{ij}\varepsilon \end{aligned} \tag{4}$$

For the welding simulation the material response is modelled by a classical time independent thermoelastoplastic material model in terms of J_2 incremental theory with a temperature dependent yield surface. The conditions during yielding are described by the associated by flow rule also known as the Prandtl-Reuss relation, Mendelson [8] and [9]

$$\dot{\varepsilon}^{pl}_{ij} = \dot{\lambda} s_{ij} \tag{5}$$

where the plastic strain increment is a function of the load parameter $\dot{\lambda}$ and the outwards normal, s_{ij}, to the von Mises yield surface is given by the following function

$$f(\sigma, \varepsilon^p_e, T) = |\sigma| - \sigma_Y(\varepsilon^p_e, T) \leq 0 \tag{6}$$

During yielding the yield condition is enforced by the consistency condition to ensure the stress conditions in the material consist with the actual yield strength of the material

$$\dot{f}(s_{ij}, \varepsilon^{pl}_e, T) = \frac{\partial f}{\partial s_{ij}}\dot{s}_{ij} + \left.\frac{\partial f}{\partial \varepsilon^{pl}_e}\right|_T \dot{\varepsilon}^{pl}_e + \left.\frac{\partial f}{\partial T}\right|_{\varepsilon^{pl}} \dot{T} = 0 \tag{7}$$

The continuum stiffness tensor can be derived by using the above information, se e.g. Tvergaard [6] and Thorborg [11]

$$\dot{\sigma}_{ij} = \left(L^{el}_{ijkl} - \beta 2\mu \frac{n_{ij}n_{kl}}{1 + \frac{E_P}{3\mu}} \right) \dot{\varepsilon}_{kl} \tag{8}$$

Clearly, this time independent formulation does not include time effects such as creep and the magnitude of the plastic strain increment is given by the load parameter $\dot{\lambda}$ which is determined by the consistency condition only. This material description is considered to be adequate for the welding simulation where the first order effect is the yield strength of the material as a function of temperature. For the heat treatment simulation, however, time is an important parameter and creep effects are considered

since creep strain rates can be relatively high at elevated temperatures. Therefore, time is considered and the creep strain rate is given by a power law creep model

$$\dot{\varepsilon}^{cr} = \dot{\varepsilon}_0 \sigma^n \tag{9}$$

which is used in the generalized expression for the creep strain

$$\dot{\varepsilon}^{cr}_{ij} = \frac{3}{2}\dot{\varepsilon}^{cr} s_{ij} \tag{10}$$

For this type of flow rule the material actually flows for all stress levels. But due to the temperature dependent material data the creep strain rate is very sensitive to the temperature level as well as the stress level. The reference strain rate is considered temperature dependent and given by an Arrhenius expression

$$\dot{\varepsilon}_0 = A \cdot exp(-\frac{Q}{RT}) \tag{11}$$

where Q is the internal energy, R the gas constant, A is a constant and T the absolute temperature.

Numerical implementation

To ensure a numerically stable and fast algorithm the constitutive models are implemented by using a backward Euler time discretization, Kleiber [14] and Sivakumar and Voyiadjis [15]. This is obtained by using the well established return mapping algorithm to enforce the yield condition on implicit time level and by that fulfilling both the fundamental equilibrium condition and the constitutive yield condition on new time level. The return mapping algorithm is scaling the deviatoric stresses to lower an elastic trial stress to a stress state that fulfills the yield condition Simo and Hughes [10]. Generally this algorithm solves a strain driven problem by assuming elastic conditions initially and applying plastic relaxation if required. To establish the numerical algorithm, Newton Raphson iterations are used to obtain the global equilibrium, and to optimize the convergence a line search is performed after solving the system to minimize the force residual Belytschko [13].

The Radial Return Mapping Algorithm The deviatoric trial stress level is calculated according to the standard elastic relation between the deviatoric stress and the deviatoric strain.

$$^n s^{trial}_{ij} = {}^{(n-1)}s_{ij} + 2\mu^n \Delta e_{ij} \tag{12}$$

where 2μ is the shear modulus of the material, Δe_{ij} the deviatoric strain increment and s_{ij} is the deviatoric stress. Notice, Δe_{ij} is found by the equilibrium iterations, i.e. $\Delta e_{ij} = \sum \Delta e^{(k)}_{ij}$
The plastic strain increment is given by the well known Prandtl-Reuss expression

$$\Delta\varepsilon^{pl}_{kl} = \frac{\Delta\gamma}{\parallel s_{rs} \parallel}\frac{\partial f}{\partial \sigma_{kl}} = \Delta\gamma \frac{s_{kl}}{\parallel s_{rs} \parallel} = \Delta\gamma\, n_{kl} \tag{13}$$

that is the flow rule composed of a load increment $\Delta\gamma$ and a direction in terms of the unit normal vector n_{kl}.

Updating fields The solution strategy described above leads to the following update of the involved fields in a typical structure of a numerical code going from one time level to the next $[t_{n-1}, t_n]$.

$$\begin{aligned} ^n\varepsilon^{pl}_{ij} &= {}^{(n-1)}\varepsilon^{pl}_{ij} + \Delta\gamma^n n_{ij} \\ ^n\varepsilon^{pl}_{e} &= {}^{(n-1)}\varepsilon^{pl}_{e} + \sqrt{\tfrac{2}{3}}\Delta\gamma \\ ^n s_{ij} &= {}^n s^{trial}_{ij} - 2\mu\Delta\gamma^n n_{ij} \end{aligned} \tag{14}$$

To calculate the size of the load parameter, the yield condition is enforced on the new time level $t = t_n$, which can be considered as a discretized application of the consistency condition in a back-ward Euler implementation of the yield condition. This leads to the following scalar equation to determine the load parameter.

$$g(\Delta\gamma) = -\sqrt{\frac{2}{3}}\sigma_Y({}^n\varepsilon_{pl}, {}^n T) + \parallel {}^n s^{trial}_{ij} \parallel -2\mu\Delta\gamma = 0 \tag{15}$$

This equation is generally non-linear and needs to be solved by an iterative solver.

The Consistent Stiffness Tensor

The consistent stiffness tensor is derived by linearizing the general stress strain expression considering plasticity

$$^n\sigma_{ij} = \kappa\delta_{ij}({}^n\varepsilon_{mm}) + 2\mu({}^n e_{ij} - \Delta\gamma^n n_{ij}) \tag{16}$$

where $\kappa = \frac{E}{3(1-2\nu)}$

Differentiating the general expression, (16), yields the following relation for the stress increment

$$ {}^{n}d\sigma_{ij} = C_{ijkl}\,{}^{n}d\varepsilon_{kl} - 2\mu\left({}^{n}n_{ij}\frac{\partial\Delta\gamma}{\partial\varepsilon_{kl}} + \Delta\gamma\frac{\partial^{n}n_{ij}}{\partial\varepsilon_{kl}}\right){}^{n}d\varepsilon_{kl} \tag{17} $$

The two partial derivatives can be derived by applying the chain rule and solving for the derivatives. More detailed presentation of these derivations can be found in Simo and Hughes [10] and Thorborg [11]. However, using the results from these derivations and inserting these in the 'original' algorithmic expression for the stress increment, (17) yield the consistent stiffness matrix used in the numerical code.

$$ d\sigma_{ij} = \left[C_{ijkl} - 2\mu\left(\frac{{}^{n}n_{ij}\,{}^{n}n_{kl}}{\left(1+\frac{\sigma'_{Y}}{3\mu}\right)} + \frac{2\mu\Delta\gamma}{\|s_{rs}\|}\right.\right. $$

$$ \left.\left.\left[\tfrac{1}{2}\left(\delta_{il}\delta_{jk}+\delta_{ik}\delta_{jl}\right) - \tfrac{1}{3}\delta_{ij}\delta_{kl} - {}^{n}n_{ij}\,{}^{n}n_{kl}\right]\right)\right] d\varepsilon_{kl} \tag{18} $$

The effective stress function for creep

Similar to the return mapping algorithm for time independent plasticity the evaluation of the creep strain rate is done by a backward Euler time discretization. This evaluation is also based on a strain driven problem where an initial elastic trial stress state is relaxed in the deviatoric system by the creep strain. The creep strain rate is given by the assumed power law behavior and to ensure numerical stability the reference stress in this expression is evaluated on implicit time level. To evaluate the stress conditions on new time level the effective stress function is implemented, Bathe [12]. This function resembles the scalar expression, (15) used to determine the load parameter for time independent plasticity.
The function can be derived in the following steps, based on the decomposition of the deviatoric strain tensor

$$ {}^{t+\Delta t}S_{ij} = {}^{t+\Delta t}S^{tr}_{ij} - 2\mu\left(\Delta e^{cr}_{ij}\right) \tag{19} $$

Insertion of the creep strain rate yields

$$ {}^{t+\Delta t}S_{ij} = {}^{t+\Delta t}S^{tr}_{ij} - 2\mu\frac{3}{2}\Delta t\dot{\varepsilon}_0\,{}^{t+\Delta t}\sigma^{n}\,{}^{t+\Delta t}S_{ij} \tag{20} $$

Collecting terms and taking the inner product

$$ {}^{t+\Delta t}S_{ij}\,{}^{t+\Delta t}S_{ij}\left(1+3\mu\Delta t\dot{\varepsilon}_0\,{}^{t+\Delta t}\sigma^{n}\right)^{2} = {}^{t+\Delta t}S^{tr}_{ij}\,{}^{t+\Delta t}S^{tr}_{ij} \tag{21} $$

Which can be rewritten to the following transcendental equation used to solve for the effective stress on new time level.

$$ f({}^{t+\Delta t}\sigma) = {}^{t+\Delta t}\sigma\left(1+3\mu\Delta t\dot{\varepsilon}_0\,{}^{t+\Delta t}\sigma^{n}\right) - {}^{t+\Delta t}\sigma^{tr} = 0 \tag{22} $$

Coupled process modelling

Generally the numerical code is based on a semi-coupled analysis where thermal results are used as input to the mechanical solver. No coupling is done from the mechanical results to the thermal solver. To model the effects of the different process-steps during the manufacturing stages of the product, the entire history of the material is modelled and results from the welding simulation and subsequent machining process are used to initialize the heat treatment calculation. To transfer residual conditions from one process step to the other all fields are evaluated in the Gauss points, the nodes and the elements are stored and restored continuously. This process is rather straightforward since the mesh is the same during the process stages.

HP valve and material data

The enmeshment of the high pressure valve is shown in figure 3 and the dimensions of the valve are listed in the table below.

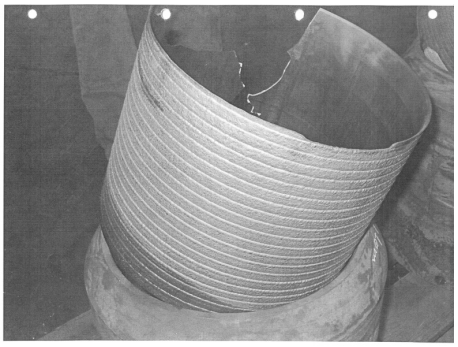


Figure 2: Stellite layer in HP valve

Measure	Dimension
Inner diameter - before welding	300mm
Inner diameter - after welding	294mm
Outer diameter	370mm
Length of valve - welded zone	250mm
Total length of valve	425mm

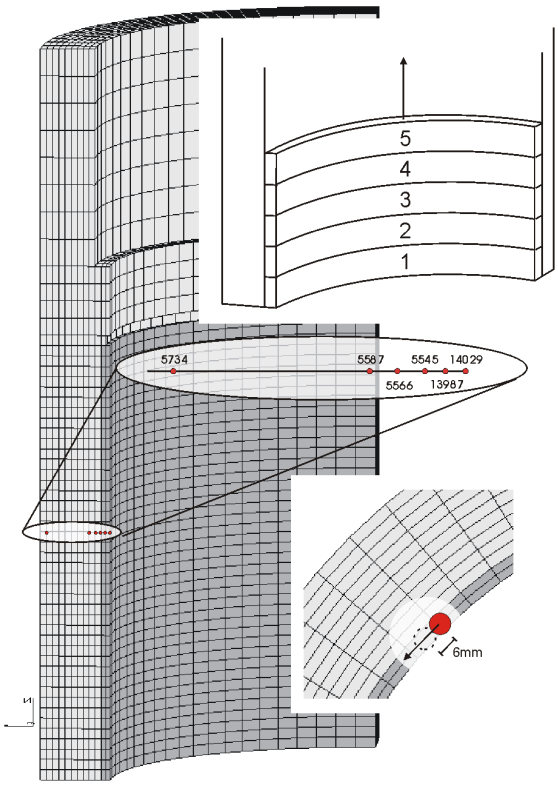
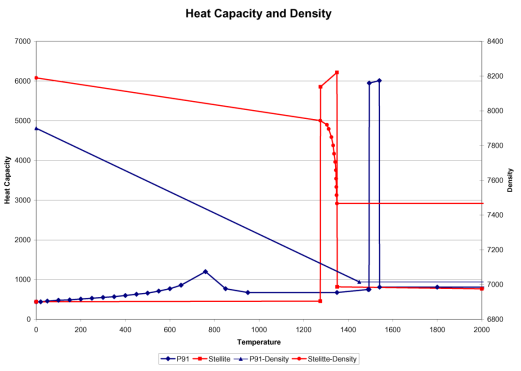
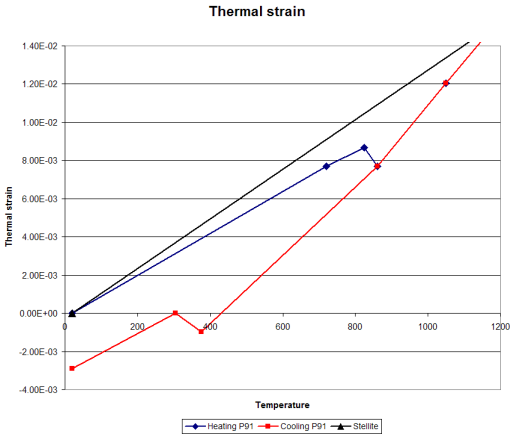
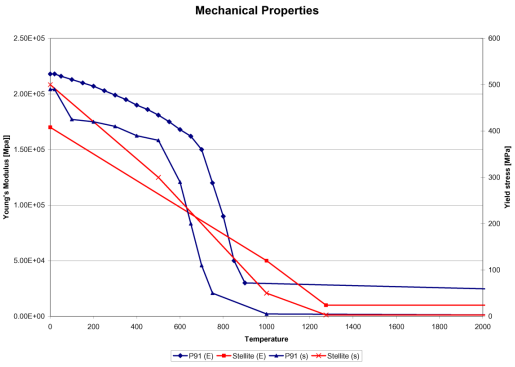


Figure 3: Geometry with nodes in the cross section and weld material modelling

The needed material data for the thermo-mechanical analysis is a full set of temperature dependent properties. The heat conductivity is in the range of 26-30$\frac{W}{m\,K}$ for P91 and 29-37$\frac{W}{m\,K}$ for stellite. Poisson's ration is 0.3 at ambient temperature and increased to 0.45 below the solidus temperature. The rest of the properties are presented in charts below.







Welding

Process description

The stellite layer is welded on the P91 base material by a PTA-process. PTA typically uses up to 400A plasma current and the heat of the plasma is mainly used to heat the powder which has very low heat transfer properties. Hence compared to a TIG welding process PTA deposition rate is some 8 times higher and the dilution with the base metal only half. Additionally the oscillation of a PTA torch during welding spreads the heat over a larger cross sectional area. By comparison with an automatic TIG process for deposition of stellite, PTA is a very low heat input process while still giving a significantly higher deposition rate. Before the welding process is started the base material is preheated to $180^{\circ}C$.

In the FE-code the adding of weld material is modelled by activating elements in the weld zone as the torch passes. These elements are preheated to $2000^{\circ}C$ and during the time it takes the torch to pass a point in the material, heat is generated by a source term in the just activated elements. The amount of generated heat inside these elements is calculated

from the total heat input minus the heat input from the preheating of the material.
To simplify the discretization of the welding process the moving heat source is modelled by adding full rings of weld material, i.e. the heat source is not moved in the direction of the circumference but only in steps in the direction of the cylinder direction, see figure 3. The time in between each ring is added is equal to the time it takes the torch for one revolution and the time the source term is switched on is equal to the time it takes the torch to pass a generic point in the weld zone.
The process parameters used for these calculations are listed in the reference, Thorborg [16].

Austenite Forming and Martensite Transformation

In this simulation the austenite forming is considered, i.e. the P91 material is assumed to be fully transformed to austenite in the regions where the temperature exceeds $980^{\circ}C$ during the welding process. This transformation changes the behavior of the thermal expansion coefficient in order to describe the transformation during austenite forming and the later martensite forming during cooling, see the thermal strain chart for illustration.

Machining

After the welding process and the cooling down of the valve the outer layer of weld material is removed by the machining process at 25000s. The removal of material, however, has only little effect on the stress distribution and is hardly visible in the curves. The fields monitored in the outermost node, 14029, goes to zero indicating that the node is removed. It should be noticed that reaction forces from tools in the machining process are not modelled, i.e. the effect from machining is mainly a redistribution of stress due to an area reduction in the cross section.

Welding Simulation - Results

Doing a full damage analysis requires a full set of results and comments on these. However, this is not within the scope of the present paper. Therefore only two contour plots of the temperature and the $\sigma_{\theta\theta}$ stress component are shown for 20s after a ring of weld material is added. At this time the base material is heated and generally the stress conditions are compressive due to thermal expansion. In the weld material, however, reaction and thermal contraction causes the stresses to be tensile. Due to this transient history, plastic yielding and thermal contraction in the weld material the residual conditions in the weld zone is generally tensile.

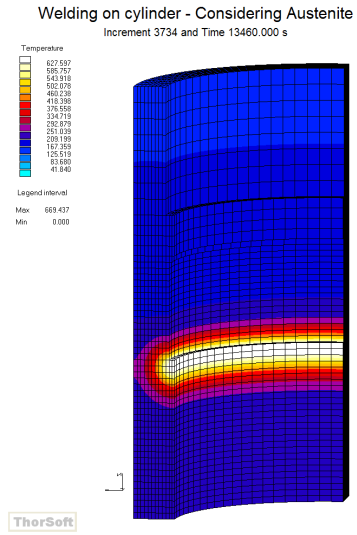


Figure 4: Temperature plot, 20s after a new weld material ring is added

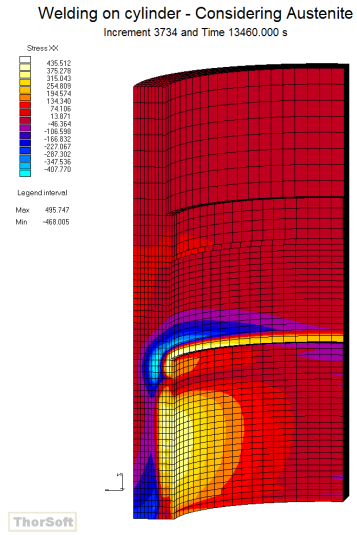


Figure 5: $\sigma_{\theta\theta}$ plot, 20s after a new weld material ring is added

Heat treatment

The heat treatment process is modelled by applying time dependent thermal boundary conditions, i.e. increasing the ambient temperature during heating, fixing the boundary temperature during the specified holding time and finally decreasing the boundary temperature to room temperature at the end.
The heat treatment process is specified to have a holding time of 4 hours. The heating and cooling rates are specified at approx. 150$^{\circ}C$ per hour.

Norton's power law

The creep behavior during the heat treatment process is described by Norton's power law. The general expression of Norton's power law is

$$\dot{\varepsilon} = A\,\sigma^{n}\,exp\left(\frac{-Q}{RT}\right) \qquad (23)$$

where $R = 8.314 \cdot 10^{-3}\frac{kJ}{molK}$. The Arrhenius expression governs the creep sensitivity to temperature.
In the heat treatment calculation three different zones/materials are considered. The first material is the weld material or stellite material, the second material is the original untransformed P91 base material and finally the third material is the martensite transformed material

Material	A	n	Q $\frac{kJ}{mol}$
Stellite	$8.2 \cdot 10^{5}$	12.95	750
Untransformed P91	$3.73 \cdot 10^{16}$	6.9	677
Transformed P91	$2.75 \cdot 10^{13}$	4.6	677

Table 6: The material data describing the three different materials

From this list of material data it is readily seen that the stellite material is the most creep resistant material even for elevated temperatures. The zone of untransformed P91 is the most creep sensitive material while the transformed P91 is far more creep resistent. These data are of course clearly seen in the numerical results presented below.

Heat treatment - Results

The heat treatment simulation is based on the residual conditions of the welding simulations and subsequent machining, i.e. all fields are initialized to the residual values to have the full history of the loading conditions in the material.
Figure 6 shows the von Mises stress distribution in the selected nodes, indicating the stress relaxation in the P91 material. Figure 7 gives the $\sigma_{\theta\theta}$ stress distribution through the cross section, at initial temperature, when 700 $^{\circ}C$ is reached, when cooling is started and when ambient temperature is reached again, respectively.

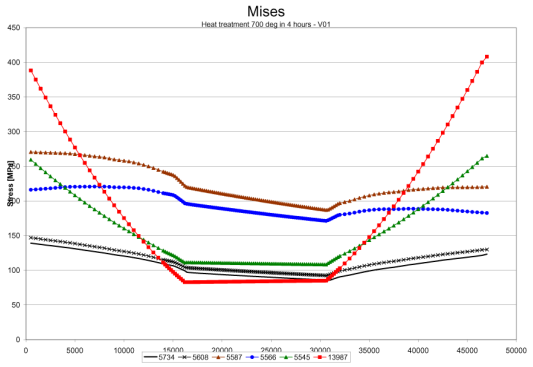


Figure 6: The von Mises stress

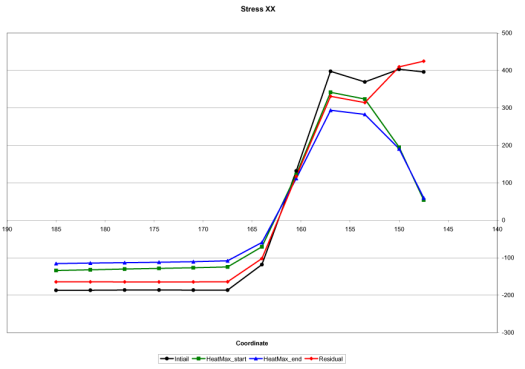


Figure 7: The $\sigma_{\theta\theta}$ stress component

Conclusion

The thermal load on the system and the subsequent developed mechanical fields are highly governed by the dynamic addition of weld material during the weld process. The two main reasons for the residual stress conditions in the material after the welding process are the difference in zero thermal strain for the base material and the weld material and the plastic yielding in the weld zone during the high temperature peaks when hot weld material is added to the system. During addition of hot filler material, the

base material is heated and starts to yield in compression and at the same time the stellite yields in tension. But during cooling both materials contract and at the residual conditions the stellite material has a high thermal contraction. All together this gives a residual condition of tensile stresses in the weld zone. The machining process removes the top layer of weld material. This geometrical change only redistributes the stresses slightly and the effect of this process stage is negligible.

During the heat treatment process the thermal load on the stellite material is decreased during reheating of the material. This is immediately seen in the stress level locally in the stellite material. However, the area of the weld material in the cross section is small compared to the base material and the influence on the P91 material is small even around the weld zone. During the stationary part of the temperature history the untransformed part of the P91 material starts to creep and the stress level generally relaxes in the P91 material. During this stationary part of the process the stress level is almost constant in the stellite material. During cooling the thermal load is restored in the system and the stresses increase again. But due to the relaxation in the P91 material the stress level in the stellite material actually increases a little bit compared to the initial conditions.

Generally, the welding process causes an inexpedient tensile stress distribution in the structure and the subsequent heat treatment process does not improve the situation, to some extent it increases the stress level in the stellite layer. In-service load calculations must be performed to clearly assert the creep behavior during service load conditions and evaluate formation of critical crack zones.

References

[1] Carslaw HS., Jaeger JC. *Conduction of Heat in Solids*. Oxford: Oxford Science Publications, 1959.

[2] Olhoff N. *Noter til styrkelaere II*. Lyngby: DTU Institut for Fastofmekanik, 1962.

[3] Hattel J. *Control volume based finite difference method – numerical modelling of thermomechanical conditions in casting and heat treatment*. Lyngby: DTU Dept. Manufact. Eng., 1993.

[4] Hattel J., Hansen PN. *A control volume based finite difference method for solving the equilibrium equations in terms of displacements*. Applied Mathematical Modelling, 1995;19(4):550-559.

[5] Hill R. *The mathematical theory of plasticity*. Oxford: Clarendon press, 1950.

[6] Tvergaard V. *Plasticity and creep in structural materials*. Lyngby: DTU Department of Mechanical Engineering, 2001.

[7] Lubliner J. *Plasticity Theory*. The Macmillan company, 1990.

[8] Mendelson A. *Plasticity: Theory and application*. The Macmillan company, 1968.

[9] Kachanov LM. *Fundamentals of the theory of plasticity*. Mir publishers, 1974.

[10] Simo JC. *Computational inelasticity*. New York: Springer Verlag, 1997.

[11] Thorborg J. *Constitutive Modelling in Thermomechanical Processes, Using The Control Volume Method on Staggered Grid*. Lyngby: Dept. Manufact. Eng., Techn. Univ. Denmark, 2001.

[12] Bathe KJ. *Finite Element Procedures*. New Jersey: Prentice-Hall, Inc., 1996.

[13] Belytschko T., Liu WK. and Moran B. *Nonlinear Finite Elements for Continua and Structures*. New York: J. Wiley & Sons, 2000.

[14] Kleiber M. *Handbook of computational solid mechanics*. New York: Springer Verlag, 1998.

[15] Sivakumar MS., Voyiadjis GZ. *A simple implicit scheme for stress response computation in plasticity models*. Madras, India: Springer Verlag, 1997.

[16] Thorborg J., Hald J., Hattel J. *Stellite failure on a P91 HP valve - failure investigation and modelling of residual stresses*. In: Proceedings of IIW Commission IX, behaviour of metals subjected to weling, Osaka, 2004.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

SIMULATION OF CAPACITOR DISCHARGE STUD WELDING

Dietmar Weiss[1] and Thomas Lauenroth
[1)] Technical University of Denmark
Department of Manufacturing Engineering and Management
2800 Lyngby
Denmark

Abstract

A mathematical model was developed for analyzing the energy balance in initial-contact mode of capacitor discharge stud welding. Due to the difficulties to determine experimentally the heat transfer in the fusion zone, measured electric values are combined with a simulation of the generation of heat and its distribution. The model takes the effects of latent heat and stud tip deformation into account. In the arc phase, it simulates the melting of the mating faces and computes the critical point in time before the joining phase must start to avoid potential failure of the joint. The model aids optimizing of the process from an energetic point of view by derivation of optimal discharge current profiles.

Keywords: Capacitor discharge stud welding, energy balance, discharge profile optimization

Nomenclature

A	area [m^2]
E	electric energy [J]
F_F	spring force [N]
H	enthalpy [J]
h	specific enthalpy [J/kg]
I	discharge current [A]
$\hat{n}$	normal vector
P	power [W]
$\dot{Q}$	heat flux [W]
Q	heat [J]
$\vec{\dot{q}}$	heat flux density [W/ m^2]
T	Temperature [ºC]
t	time [s]
U	voltage [V]
V	volume [m^3]
λ	thermal conductivity [W/(m K)]
ρ	density [kg/m^3]

Introduction

Capacitor discharge stud welding (CDSW) is an efficient arc process for welding of fasteners with diameters of up to 10 mm on various kinds of metals. Energy stored in the capacitor is discharged directly trough stud and workpiece. Thereby the mating faces of stud and plate are molten by resistance and arc heating. The process is widely used because of economic and quality-related advantages [1]. The equipment is inexpensive and the process can be applied to most relevant combinations of ferrous or non-ferrous metal. In Fig. 1 and 2, sketches of the set-up of CDSW in initial-contact mode and of the process phases

- ignition phase
- arc phase
- joining phase

are given.

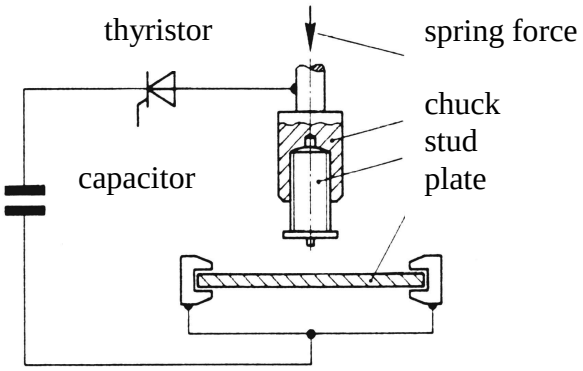


Figure 1: Sketch of capacitor discharge stud welding

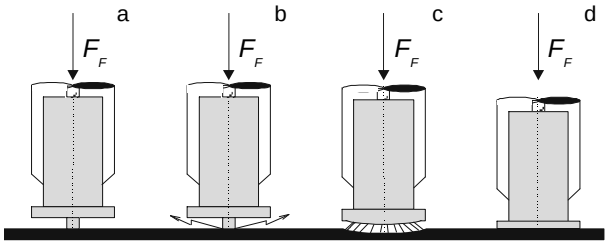


Figure 2: Initial-contact mode: a) initial position, b) ignition, c) arc phase, d) joining phase

In comparison to other welding processes, the plate will not be deformed or affected in strength by drilling of holes. Due to the short welding time, shielding by an inert gas is unnecessary and the maximum temperature on the reverse plate side is relatively low. For instance, fasteners with a diameter of 6 mm can be welded on 1 mm plastic coated steel sheets without marking the coating on the reverse plate side. However, the explosive-like sound with a level of up to 140 dB and the difficulties to reproduce reliably an acceptable joint quality are still critical issues. Figure 3 shows exemplarily a typical defect where the energy remaining after the ignition phase was insufficient. Consequently, the mating stud and plate faces were only partially melted causing a lack of fusion.

Therefore, current research is focused on optimization of discharge profiles with the intention of 1) decreasing the sound level and 2) reducing the capacitor size or increasing the operating time of portable chucks, respectively.

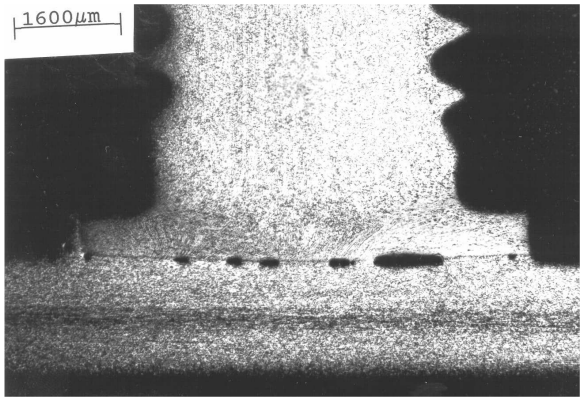


Figure 3: Lack of fusion due to partly melting of mating faces of stud and plate

The primary problem in choosing process parameters avoiding early exhaust of the capacitor are the manufacturing tolerances of the studs. For instance, the standard EN 13918 allows variation of the tip diameter of M8 studs of 0.75 ± 0.08mm. The influence of such a variation of the stud tip dimensions or of the roughness on both thermal activation and subsequent arc phase is difficult to compensate by modifying process parameter sets manually in a trial-and-error manner. Thus, further development of stud welding chucks requires the availability of real-time algorithms controlling the conversion of electric energy to heat in the fusion zone. The performance of measurements needed for formulation and validation of such control models is challenging due to the short-time character of the process. Electric process parameters and the sound level are uncomplicated to monitor. But, they only allow very limited conclusions regarding melting, evaporation or spattering. Because of the significant influence of these thermal effects on the success of electrode melting, their understanding is the key to optimization of the process. However, the dynamics of CDSW and the high temperatures make it practical impossible to measure heat transfer related effects with a sufficient precision in the thin gap between stud and plate. Besides, observation of the liquid region cannot be performed in the joining phase.

Therefore, the experimental work should be supported by numerical modeling of the generation of heat and its dissipation. The scope of the investigation is focused on minimizing the energy needed for ignition of the arc and melting of the electrode faces. Several models of stud welding have been developed [4,5,6], but they are not

applicable to the arc phase of the initial-gap mode considered here. For this mode, the thermal processes have only been simulated in the ignition phase [7]. Therefore, a model should be proposed which allows the prediction of temperature distribution and of heat fluxes to plate and stud as function of measured process parameters. The simulation code takes the effects of evaporation, deformation of the molten tip and movement of the stud into consideration. Results are the enthalpy necessary for melting and disintegration of the stud tip, the enthalpy required for melting of the electrodes over the cross-sectional area of the joint and the critical point in time before the joining phase should start in order to avoid potential failure of the joint.

Modeling

An example of the experimental data available is given in Fig. 4. Discharge current and voltage measured reflect conversion of electric power to heat as an integral value defined over a domain containing plate, stud and fusion zone. Due to the space-independence of the input data, it seams to be appropriate that a lumped model is stated based on the following assumptions:

- transient one-dimensional heat transfer problem in axial extension of the stud
- plane and parallel electrode faces with an area less or equal the cross-sectional area of the stud
- constant density in all states, all other thermo-physical properties are temperature-dependent
- uniform distribution of heat flux on plate and stud faces in radial direction - based on the observation of a constant thickness of the heat affected zone in radial direction
- consideration of the local displacement of liquid at the arc spot by an increased mean thermal conductivity of the molten metal in axial direction
- ratio of heat flux to workpiece and to stud equals one
- no heat generation in joining phase because electric resistance is vanishing
- similar material combination

The electric power is an arbitrary function of time $P = f(t)$ which can be modified in order to adjust the transient heat generation. The loss of mass by spattering as one of the factors determining the thickness of the molten layer will be a tuning parameter of the model. The governing equations are formulated on the domain $\Omega = (z_{\min}, z_{\max})$ where $z_{\min}$ and $z_{\max}$ denote the axial coordinates of the reverse side of the plate and the stud end.

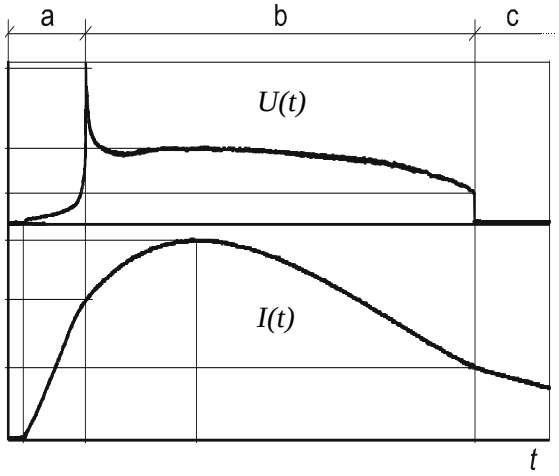


Figure 4: Voltage and discharge current versus time in a) ignition, b) arc, c) joining phase

$$\int_V \rho \frac{\partial h}{\partial t}\, dV = \int_A \vec{q} \cdot (-\hat{n})\, dA + \int_V \dot{q}_{vol}\, dV \qquad (1)$$

$$\vec{q} = -\lambda \operatorname{grad} T \qquad (2)$$

$$\dot{q}_{vol} = \frac{\dot{Q}_{inp}}{V_{gap}} \qquad (3)$$

$$\dot{Q}_{inp}(t) = \eta\, U(t)\, I(t) \qquad (4)$$

V denotes the volume, ρ the density, h the specific enthalpy, t the temporal variable, $\vec{q}$ the heat flux density, $\hat{n}$ the outward pointing normal, A the area, $\dot{q}_{vol}$ the volume-specific source, λ the thermal conductivity, T the temperature, $\dot{Q}_{inp}$ the net heat input, η the thermal efficiency, U the voltage and

I the discharge current. The boundary conditions on the boundary $\partial\Omega$ are of Dirichlet type and equal the initial temperature T_0 = 20°C. The transfer problem described by equations (1)-(4) is discretized employing a cell-centered finite volume method.

In the ignition phase, $\dot{Q}_{inp}$ is characterized by a contact and a material resistance of the stud tip, see [7] for details. In the arc phase, the heat available for melting of stud and plate faces is the difference between the electric power of the arc converted to heat and the heat transferred to the solid metal by conduction and to the environment by radiation, convection and evaporation:

$$\dot{Q}_{melt}(t) = P - \dot{Q}_{out} - \dot{Q}_{loss} \qquad (5)$$

$\dot{Q}_{melt}$ is mainly determined by discharge profile and cooling conditions and subject of rapid temporal variations. The application of Equ. 5 allows to derive the desired information about the thermal state of the fusion zone. $\dot{Q}_{out}(t)$ is calculated according to Equ. 2 based on the gradient of the actual temperature field at the liquid-solid interfaces.

Results

The simulation results reflect the thermal processes in arc and joining phase as demonstrated in Fig. 5 for a steel fastener M8, S235 with tip length l_T = 0.82 mm, tip diameter D_T = 0.83 mm, capacity C = 66 mF, no-load voltage U_C = 215 V, inductivity L_K = 6 µH, contact resistance R_K = 9.5 mΩ and spring force F_F = 130 N. The heat flux available for melting of the electrodes $\dot{Q}_{melt}$ is represented as the difference between $\dot{Q}_{inp}$ and $\dot{Q}_{out}$. At the begin of melting, the arc effect is limited to the small cross-sectional area of the stud tip with a radius of D_T. The net heat generation of $\dot{Q}_{inp}$ between 100 and 150 kW is relatively high and causes an initial heat flux density $\dot{q}_{out}$ greater than 100 kW/mm². The reduction of $\dot{Q}_{melt}$ by heat conduction can be neglected because of the small cross-sectional area of the tip. The area of arc activity is increasing until a net heat input of 80 Ws is deposited in the fusion zone after 0.7 ms which corresponds to a thermally activated area of 50 mm² and a heat flux density $\dot{q}_{out}$ of 1.6 kW/mm². The positive $\dot{Q}_{melt}$ allows heating and partial melting of the electrodes. At this point, an amount of 160 kW of the generated $\dot{Q}_{inp}$ = 200 kW is transferred to the colder electrode regions. Thus, the melting heat available is $\dot{Q}_{melt}$ (t = 0,7 ms) = 40 kW. It drops rapidly with falling discharge current and is zero before the end of the melting phase at t = 1.4 ms. Having an absolute value of 90 kW, the flow in and out of the process zone is in balance as indicated as point (a) in Fig. 5. The still decreasing melting heat becomes negative before the transition to the joining phase. At t = 1.6 ms, a power of 45 kW is converted to heat, but 71 kW are transferred to the solid parts. Consequently, the liquid layer cools down in spite the fact that a substantial amount of heat is still transferred to the fusion zone. The plot of the computed temperatures between t = 0.9 ms and t = 1.6 ms over the axial coordinate in Fig. 6 confirms this. The result shows that the faces of plate and stud are fully activated at t = 0.87 ms and that the mean thickness of the liquid layer on both sides is 50 µm. The velocity of the movement of the melting front decreases with falling melting power at the end of the process, Fig. 5 and 6.

Even with negative $\dot{Q}_{melt}$, the melting of the electrode faces continues slowly, but enthalpy and the mean temperature of the liquid layer are dropping. Thereby, the crystallization of the molten metal will start before reaching the point of transition to the joining phase. If the mass of the molten metal is reduced to a value below the critical mass for joining, the risk of a failure of the operation is given. Therefore, the setting of the process parameters has to secure that the joining phase ends before or at the point in time when $\dot{Q}_{melt}$ = 0 as shown in Fig. 7a) for a medium inductivity corresponding to a moderate profile of the discharge current. For smaller inductivities, mechanical limits of the process cause the risky transition to joining after reaching $\dot{Q}_{melt}$ = 0, Fig. 7b). Positioning of a transition point after the process end is theoretically possible when using very high inductivities in combination with high no-load voltages, Fig. 7c).

The quality criterion corresponding to a discharge profile according to Fig. 7a) ensures that a maximum mass of molten metal with high

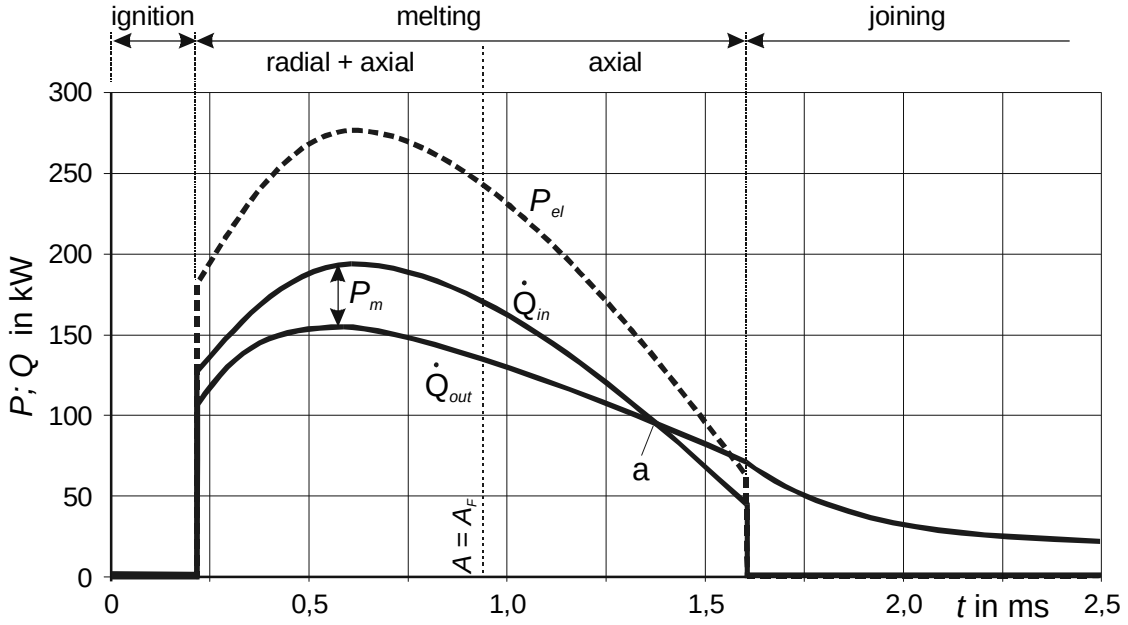


Figure 5: Net heat generation $\dot{Q}_{inp}$, heat flux to solid material $\dot{Q}_{out}$ and melting heat $\dot{Q}_{melt} = P_m$ versus time, (a) indicates point where $\dot{Q}_{melt} = 0$

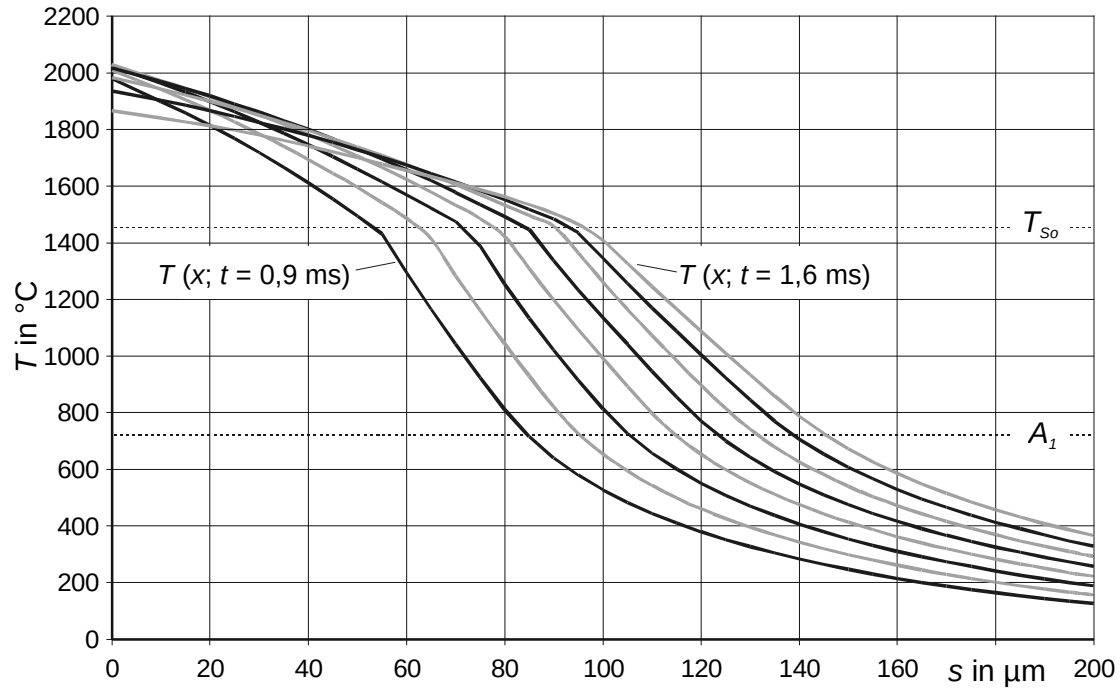


Figure 6: Axial temperature distribution versus time in arc phase

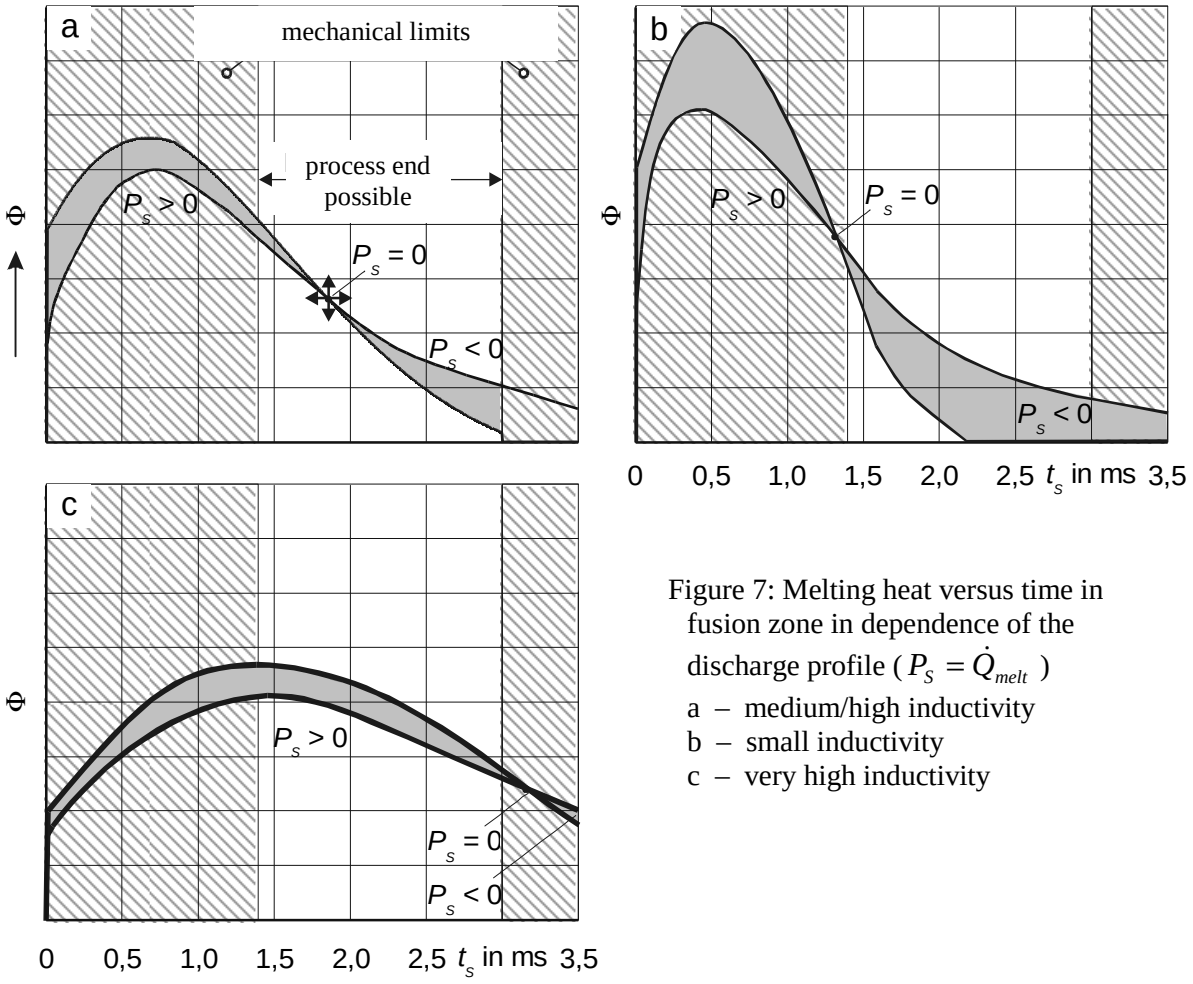


Figure 7: Melting heat versus time in fusion zone in dependence of the discharge profile ($P_S = \dot{Q}_{melt}$)
a – medium/high inductivity
b – small inductivity
c – very high inductivity

enthalpy is available. For example, for a mild steel M8 stud it is fulfilled if the maximum inductivity is 10 μH and the arc phase is shorter than 2 ms can be guaranteed with a spring force of 80 N. With the transition to the joining phase, the net heat input $\dot{Q}_{inp}$ vanishes. In the example considered, $\dot{Q}_{out}$ dominates the process if $t > 1.6$ ms, Fig. 8. Starting with around 70 kW, $\dot{Q}_{out}$ drops with falling temperature difference between molten layer and electrodes. The cooling of the liquid accelerates and its mean temperature drops. If the melting temperature is reached, crystallization starts and the two crystallization fronts join at the center of the process zone. The theoretical crystallization time is 2.3 ms (Fig. 9a), the time drops to 0.62 ms if a realistic mass loss by spattering of 50% is assumed (Fig. 9b). The simulation results show that the enthalpy brought into fusion zone and base metal can be reduced by modification of the discharge profile. This is advantageous when manufacturing thin coated sheets or panels. Further development should allow manufacturing even thinner sheets which will reduce costs essentially and make CDSW more attractive.

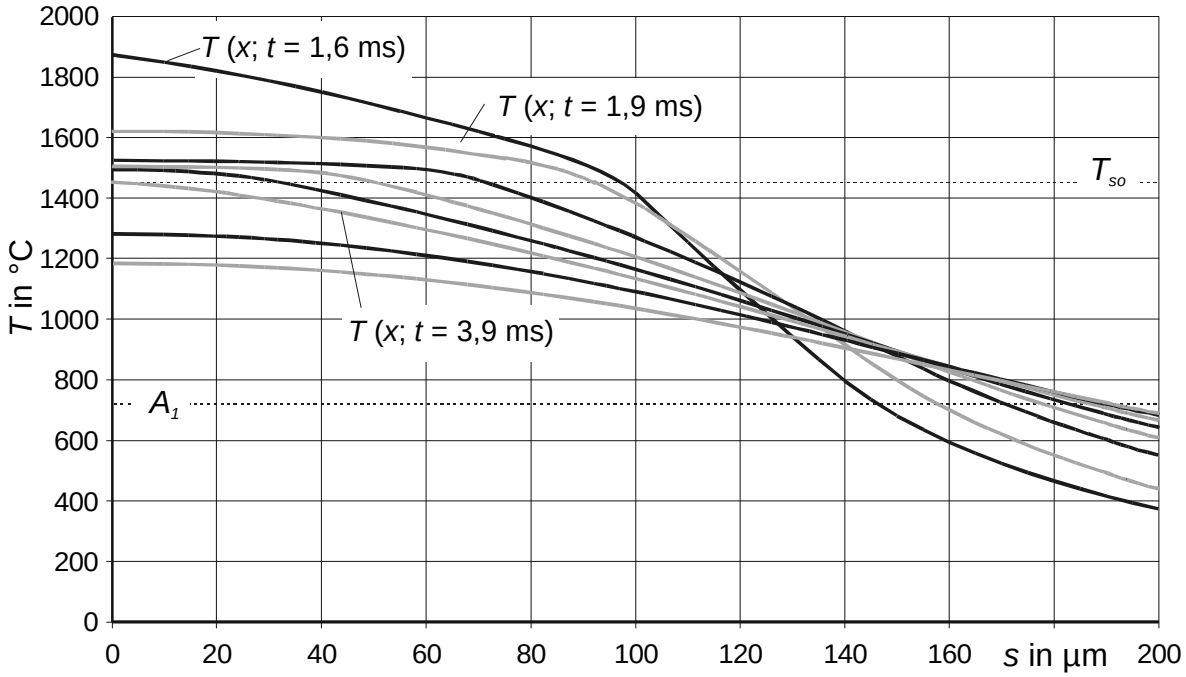


Figure 8: Axial temperature distribution versus time in joining phase

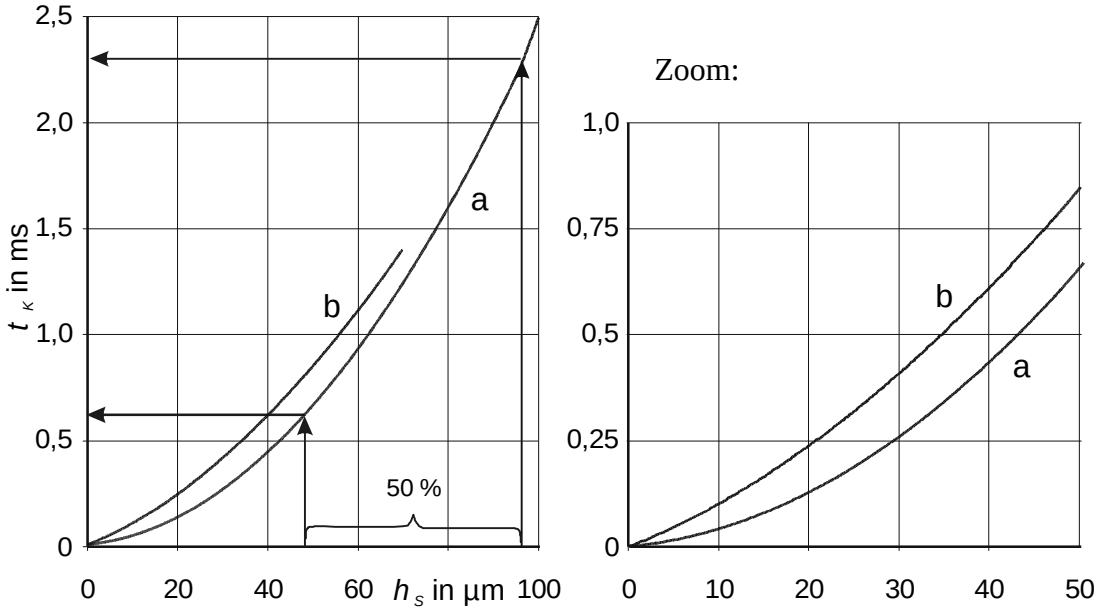


Figure 9: Calculated crystallization time versus thickness of liquid layer
a – M8, S235, C = 66 mF, U_C = 215 V, L_K = 6 µH, R_K = 9,5 mΩ, F_F = 130 N,
b – M8, S235, C = 66 mF, U_C = 185 V, L_K = 12 µH, R_K = 10,5 mΩ, F_F = 35 N

Conclusions

The problem of energy conversion and heat dissipation in the ignition and arc phase of capacitor discharge stud welding is simplified as a one-dimensional heat transfer problem. The effects of evaporation and convective transfer are considered by modified heat sources and effective material properties. Measurements of current and voltage with a high resolution in time have been combined with the analysis of the energy balance over the gap region between stud and workpiece. The resulting model allows finding optimal transition points from arc to joining phase and the energy necessary for sufficient melting of the electrode faces before joining. The method is applicable to different material combinations and stud dimensions.

References

[1] ISO 14555:1998, *Welding - Arc stud welding of metallic materials*

[2] EN ISO 13918:1998, *Welding - Studs and ceramic ferrules for arc stud welding*

[3] ASM Handbook, *Welding, Brazing and Soldering*, Vol.6, 1997,ASM International, p. 222-23

[4] Schmitt, K. G.: *Untersuchungen zur Prozessanalyse und prozess-begleitenden Qualitätskontrolle beim Bolzenschweißen mit Spitzenzündung*, Dissertation, Univ. - Gesamtschule Paderborn, Germany, 1983

[5] Stache, K.: *Prozessanalyse und Maßnahmen zur Qualitätssicherung beim Bolzenschweißen mit Spitzenzündung*, Dissertation, Technische Universität Clausthal, Germany 1997

[6] Kaleko, D. M.; Lebedev, V. K. ; Chvertko, N. A.: *Processes of Welding using the Arc Discharge of the Capacitor*, Welding and Surfacing Reviews, Volume 13, Part 2, E. O. Paton Electric Welding Institute, Ukrainian Academy of Sciences, Kiev, Harwood Academic Publishers, 1999

[7] Weiss, D.; Franz, U., Mahrle, A.; Schmidt, J.: *Modelling of capacitor discharge stud welding*. 10th TWI Conference on Computer Technology in Welding, Copenhagen 2000

[8] Lauenroth, T.: *Process analysis of capacitor discharge stud welding with tip ignition*, Ph.D. thesis, Technical University of Clausthal, Germany, 2002

[9] Weiss, D.: *Model for numerical simulation of transient heat transfer problems in CDSW*. Unpublished simulation code, Copenhagen 2000

[10] Rohloff H.; Zastera, A.: *Physikalische Eigenschaften gebräuchlicher Stähle: Daten für Hersteller und Anwender*, VDE-Institut für Angewandte Forschung GmbH Stahleisen, Düsseldorf, 1996

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

Process Optimization and Diagnostics

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

Conditions for Intelligent Process Diagnostics and Optimization in the Future.

Erik Dahlquist and Janice Dhak, Malardalen University, Vasteras, Sweden

ABSTRACT

In the paper the conditions for bringing the research to practical use in process industry is discussed. A tight cooperation with the users in process industry is needed. Uncertainties and strategies to handle these have to be addressed. Integration of diagnostics, optimization and decision support systems is needed.

1. INTRODUCTION

Two thousand years ago a Greek scientist and philosopher claimed that "we now know everything, and we can skip further research activities". A hundred years ago the president for the US patent office claimed "all possible inventions are done, and we can now close the patent office". Today we sometimes feel we have already developed all tools necessary for process monitoring and optimization, but as very few are in real usage at mills, we can be sure there is much more to be done.

The emphasis thus will have to be to develop functions that can bring the applications into productive use. We need robustness and integration between different tools. If we cannot rely on the process signals we can not rely on the control, and automated optimization will be irrelevant.

We also must have a very good dialogue between those who are to use the tools in the plants. Today even very good tools are not used just because the "driver" at the mill has retired or moved somewhere else, and then no one else bothers to use the tool.

To come around this we need intuitive designs of process displays, so that the operator and engineers will understand the meaning of the functions easily. It shall not be needed to be a PhD to use the tools. Very often we as "theoretical experts" want to have all flexibility in the world, which makes it difficult even for ourselves to know the meaning of the alternatives half a year after the algorithms were created.

In this paper we discuss the conditions needed from the demand side. What functions do we want to have? Secondly we discuss intuitive design of process displays and how to communicate in an efficient way between operators and system. After this we make a technical overview over the hierarchy of an "ideal" system with Intelligent Process Diagnostics and Process Optimization. This is the direction for the future development we believe we should strive for.

2. PROCESS OPTIMIZATION HIERARCHY

If we start with the functions, we need to monitor the process performance. Information from the process needs to be handled in some automatic way to extract important information from "noise". When we know the status of the process and combine with what we want to produce, we give the incentive for dynamic process optimization. Still, there will always be uncertainties in how the process will perform in the future. Thus we need tools for predicting the process performance as well as possible changes in the production planning during the next day and other time horizons. Finally we need to have good procedures for communication between the

developer of the system, the SW vendor and the users of the system. This must be done

The hierarchy of the functions needed for a process optimization system can be given by the figure below.

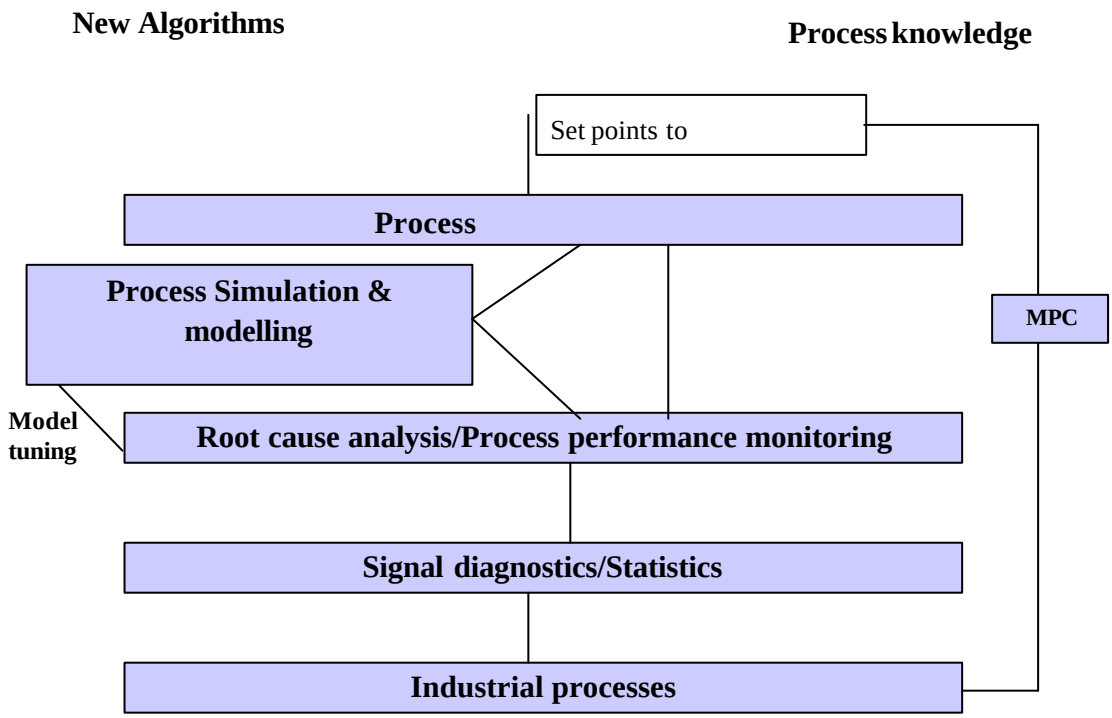


Figure 1. Process Optimization hierarchy

3. PROCESS PERFORMANCE MONITORING

Process performance monitoring is done by performing data reconciliation of the sensors and equipment in the process. Physical models are in many cases good to use as a tool for data reconciliation. All known physical relations as well as configuration of the network connecting different equipment and sensors to each other is collected. The mathematical models of the equipment are adapted or tuned to real plant data. Soft sensors can be achieved giving performance indices for different process equipments, as well as tools for data reconciliation. A steady state model is good for identifying mass and energy balances, but may have to be complemented by a dynamic model to get correlations between events and responses with some time delay in-between. Trending of fault development with moving window techniques will give a possibility to find faults in noisy processes. The time plot will also detect trends in fault development that may not be seen in the steady state calculations, if different filters are used.

4. PROCESS OPTIMIZATION

The process optimization has to be done in different time perspectives. To use the optimized set points achieved by dynamic optimization we need tools for predicting the process performance. Otherwise we may drive the process to unstable conditions. It is important to develop strategies for how to handle any kind of uncertainties. It may be that we have to reschedule production because a prioritized customer demands a fast delivery or because some unexpected event takes place in the process and makes

the old optimal operation schedule obsolete. Then it is important to get this delivered and produced without more disturbances then necessary for other customers. We have to handle other uncertainties like faulty or uncertain measurements as well as possible changes in the production planning due to problems fulfilling special quality demands. Different equipment will have different reliability, and thus should also be handled in different ways with respect to uncertainties. If a sensor normally is unreliable we should first try to move it as the position may be bad. A consistency sensor for instance may give faulty measurements if sitting where the flow is very turbulent just after a pump, but may give correct results just a few meters further down the line. To replace the sensor with a better type is a second alternative. It may also be possible to correlate it to other signals and measurements, to make the reading more reliable. This can be done by using "data reconciliation" methods. A number of articles describes methods for this, eg : Crowe et al (1983), Karlsson et al (2003), Narasimhanet al (2002), Romagnoli et al (2000), Weiss et al (1996).

When it comes to the actual optimization there are many different type of procedures and methods available. We can try to formulate the optimization problem directly in an objective function and with suitable constraints, and then use one of the standard optimization algorithms available. One way here can be to use a tool set like Tomlab which makes it possible to principally formulate your problem once, and then link to many different solvers (more than 70 today), to test which one is best suitable (Dhak et al 2004). This is the method the mathematician would prefer.

Other ways can be to use Simulated annealing or using Genetic algorithms, which are more "engineering type" of algorithms.

Both types have there benefits and draw backs.

What will be the task for the future is to use these methods, but primarily to try to identify good ways to formulate the optimization problem in a good way. By making the right simplifications, the problem may be very much faster to solve, than originally. Strategies will have to be identified for how to include uncertainties. If we want to optimize a cogeneration plant and an energy system with both heat and power production, and with several production units, the weather forecast is one factor to consider. Other factors are energy prices, taxes, fuel price, quality of fuel (how frequent do you have to steam blow to keep the precipitation on heat exchanger surfaces under control), return district heating water temperature, heat storage and others. If we just make some assumptions that the price will be this, the temperature that etc, and the assumptions were wrong, the principally optimal solution may very well be a disaster from a supply point of view, with economical losses in relation to only manual control. How do we encounter the new information being available to the operators, and making them adjust the schedules? A good attempt to do as much as possible of this has been done by Persson et al (2003) at a Swedish pulp mill, Gruvon, where optimization of two integrated recovery systems and mills are optimized with respect to production capacity and sulfur balance.

These strategies first means we will have to register all possible uncertainties. How often do different events take place? How much differs weather forecasts from future facts? Are the predictions better or worse during certain weather conditions? Do we know of future shut downs in other power plants that may trigger the electricity price? Where do we get that information, and could we get them automatically into the optimization system? These questions need to be introduced into the optimization system to give "risk values". When the prediction horizon is very uncertain, it may be better to operate the system in a very conservative way, to avoid major mistakes. On the other

hand, if we can get a better understanding of the real risk, we can minimize the uncertainties by time. This will be a real challenge for the researchers in the near future to find these strategies for handling the uncertainties, but when it is done optimization can really take off in real applications on a large scale.

Another application of optimization is the on-line use of e.g. MPC, model predictive control. This means multivariate control, where the interaction between several control loops is coordinated by some kind of model. The principles of this technology is decades old (see e.g. Morari et al 1980), but there have been practical problems to get robust systems, and thus the usage is still not very common. If we make the checking of signals, can produce robust models and can handle uncertainties in a good way, the usage will be very common. I should say if, but when, because a lot of work is going on in the field, and sooner or later we will be there. A major limitation so far has been computer calculation speed, as it takes to long to calculate. Another obstacle is to get the correct feed back to the control system, as the variables we really want to optimize often can not be measure directly in the process. By using different types of soft sensors still we are getting more and more quality sensors for this purpose (e.g. Pettersson ,1998). A lot of small steps are taken all the time, and as each step is small it is difficult to notice the strong movement, but it is there!

It is also of major interest to combine the diagnostics and process schedule with a decision support system. This can preferably be a adaptive Bayesian net type of tool, where correlations between events and resulting faults can be organized in a systematic way. The net with the relations then can be tuned to get the weights of different influences from real plant data. Here we have the same need for evaluation of the reliability of the measurements and other information from the process, to avoid mistakes. On the other hand we also here have a means to present the probability for an actual situation, if there is a sensor fault, poor performance of the process or something else. This evaluation of the probability for one or the other is adjusted by feeding back information about what was the root cause of a problem with similar characteristics earlier. Examples of papers describing this type of systems are presented in e.g. Jensen (2001) or Jensen et al (2002).

5. SYSTEM DEVELOPMENT

The most important factor for getting anything good out of the advanced functions is good procedures for *communication between the developer of the system, the SW vendor and the users of the system.* This must be done all the way from starting planning to long term operations. Otherwise the whole project will be a waste of money and time. Almost all projects will fail due to a miss in this step. The operators have too many other things to do. They do not see the advantages with the new system or even fear they will loose their jobs if it is good enough. Or they just find the system to complex to really understand how to use. Or small faults in the technology make them tired of the whole thing, if they are not engaged in solving the small problems. And for this they need a strong support from their managers. If they know their managers find this important, they will be more positive, than if the work only results in less production and thus less bonus money! So short term interests have to be balanced by long term benefits, and this must be clearly stated by the management. The best is if all levels from the president to the operators are involved and informed from the beginning and on a regular basis. Continuous information will keep the project going even if there are some back lashes now and then, something that is very difficult to avoid when advanced functions are to be implemented.

In the project we thus need to identify what functions should be included. It is better to have the complete line from process signals

to full blown decision support and optimized production for a small section of the plant first. This will show that the method is working. If we take a significant step in the whole plant, but never reach a full functional delivery it will just make the operators disappointed. On the other hand it may be that you do not see the advantages for only this small part, as they pay back is coming first when the whole installation is done. It is thus very important to go through potential savings from the beginning and to analyze where the real savings are to be found. If we can solve one problem the principles has been proven and we can take a second step.

To get an intelligent decision support we also need feed back from the operators and process engineers about real faults and problems in the process. We need feed back when a problem has been fixed, and the analysis tool needs to understand that a fault has been fixed. To get this into an automated system is a real challenge, but necessary long term. If we can get this to work properly the advise from the system will become much more reliable, which will give the operators confidence in the system. If we also can convince the operators that they will not risk their jobs just because the functions are better, they hopefully will see this in a positive way. A modern good car has a lot of new functions which makes it easier and safer to drive, but still the driver is needed! And it is not more boring to drive. Now you can try to minimize your energy consumption by watching the liter/km fuel consumption by driving in different ways. This is possible with the car computer, which has come the last years!

6. CONCLUSIONS

A lot of activities are going on all over the world on developing different features for diagnostics, optimization, control, MPC and decision support. Sooner or later these will become robust enough to be possible to use on a larger scale in process industry. What is needed is to integrate all the different functions, as they are depending on each other. Strategies for how to include uncertainties in the optimization and good decision tools for the operators will be a future research focus to get robust systems that will actually be used by the mill people. This also includes working together with the users to get input on what has to be included, and to discuss the usability, but also to add on features to keep robusticity.

REFERENCES

Crowe, C. M. et-al, "Reconciliation of process flow rates by matrix projection. Part I: Linear case", Kluwer, November 1983.

Dhak J., Dahlquist E., Holmström K., Ruiz J., Belle J.: Developing a Generic Method for Paper Mill Optimization. Control Systems 2004, Quebec City, Canada June 14-17. Conference proceedings.

Hess T. (2000): Process optimization with dynamic modeling offers big benefits, I&CS,August, p 43-48.

Jensen, F., Kjærulff, U., Lang, M., and Madsen, A. L. (2002). HUGIN - The Tool for Bayesian Networks and Influence Diagr ams. In First European Workshop on Probabilistic Graphical Models, pp. 212–21.

Jensen, F. V. (2001). Bayesian Networks and Decision Graphs. Springer-Verlag.

Karlsson C., Dahlquist E., "Process and sensor diagnostics - Data reconciliation for a flue gas channel", Värmeforsk Service AB, 2003, (in Swedish).

Morari M, Stephanopolous G and Arkun Y: Studies in the synthesis of control structures for chemical processes. Part 1: formulation of the problem.Process decomposition and the classification of the control task. Analysis of the optimizing control structures. American institute of Chemical Engineering Journal , 26(2), 220-232.1980.

Narasimhan S; Jordache C., "Data Reconciliation & Gross Error Detection, An Intelligent Use of Process Data", Gulf Publishing Company, Houston, USA, 2000.

Romagnoli, J. A; Sanchez, C. S., "Process systems engineering volume 2, Data Processing and Reconciliation for Chemical Process Operations", Academic Press, San Diego, USA, 2000.

Persson U, Ledung L, Lindberg T, Pettersson J, Sahlin P-O and Lindberg Å: "On-line Optimization of Pulp & Paper Production", in proceedings from TAPPI conference in Atlanta, 2003.

Pettersson J. (1998): On Model Based Estimation of Quality Variables for Paper Manufacturing. Tech LicThesis , KTH

Weidl G, Madsen A, Dahlquist E:Object Oriented Bayesian Networks for Industrial Process Operation. SIMS 2003, Scandinavian Simulation Society,Vasteras, Sweden Sept 18-19 2003.

Weiss G.H., Romagnoli J.A., Islam K.A., "Data reconciliation - an industrial case study", Computers and Chemical Engineering, Pergamon 1996, vol 20, issue 12, pages 1441-1449.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

DATA RECONCILIATION FOR REAL-TIME OPTIMIZATION FOR AN INDUSTRIAL COKE-OVEN-GAS PURIFICATION PROCESS

Richard Faber, Bo Li, Pu Li, Günter Wozny
Institute of Process and Plant Technology
Technical University of Berlin
10623 Berlin
Germany

Abstract

The *ammonia hydrogen sulfide circulation scrubbing* is a common industrial process for coke-oven-gas purification. We used a rigorous rate-based model to describe this reactive absorption and desorption process. To increase the accuracy of the model, we estimated several process parameters using a sequential parameter estimation approach. Data reconciliation was performed based on simple component balances to achieve model consistent data and identify measurement biases. The model was then validated online on a pilot plant by connecting the estimation package through the process control system. Based on the online measured data, operating cost minimization was performed and the computed optimal control variables realized in real-time. A satisfactory agreement between measured data and optimization was achieved.
Keywords: online optimization, model-based parameter estimation, data reconciliation

Nomenclature

$\bar{a}$	[m²/m³]	specific interfacial area
C		flow parameter
c		regression coefficient
f		objective function
g		set of equality constraints
h		set of inequality constraints
k	[m/s]	mass transfer coefficient
k	[1/s]	reaction velocity constant
K		equilibrium constant
$\dot{M}$	[kg/s]	mass flow
R		reflux rate
T	[K]	temperature
$\dot{V}$	[l/h]	volume flow rate
x		set of variables
x	[g/l]	concentration
α		regression coefficient
$\boldsymbol{\Theta}$		set of parameters to be estimated
σ		standard deviation

Introduction

Model-based optimization has been widely used to exploit economical and environmental potentials of industrial processes. For model-based online optimization the accuracy of the process model as well as the quality of the measurement data is crucial for the result of the optimization. However in most cases there exist discrepancies between the model and the real plant and the measurements are contaminated with measurement errors.

Most process models have parameters which have to be estimated from measurement data. To improve the accuracy of the model these parameters have to be estimated with measurement data taken directly from the plant. Therefore measurement data not only affects the quality of the optimization directly but also the quality of the estimated process model. However the measurement data used for model updating do contain measurement errors. Therefore data reconciliation has to be performed in order to identify and rectify measurement errors.

This work concerns with online optimization with economic objective and ecological restrictions for reactive coke-oven-gas purification. The *ammonia hydrogen sulphide circulation scrubbing* is a common process to perform this task. It selectively removes NH_3, HCN and H_2S while simultaneously suppresses the absorption of CO_2 with a combined reactive absorption/desorption cycle process. Optimization of this coke-oven-gas purification process is motivated by the following two facts. First, rigorous treatment of the sour gas gains more and more attention due to the increasingly stringent environmental regulations. Second, reduction of operation costs and separation of the gas into by-products increase the economic potential of this industrial process.

Due to the reactive absorption and desorption, we use a rate-based model with enhancement factors to account for the enhanced mass transfer. The model consists of a large-scale highly nonlinear equation system with parameters to be estimated from measurement data. In spite of the rigorous description a model validation with measurement data from a pilot plant indicated deviations in the range of 20-30% [4].

To improve the model accuracy, we estimated several parameters based on measurement data taken from a pilot plant with a model-based parameter estimation technique. The large-scale model and the large number of data sets used lead to a difficult estimation problem. A new three-stage decomposition strategy was proposed to solve this problem [3]. The developed parameter estimation approach is able to handle large-scale systems with multiple sets of measurement data. Due to the reduction of the size of the problem it is possible to use standard optimization software to solve the problem. However the quality of the updated model strongly depends on the quality of the measurement data.

Measurement data usually contain stochastic errors, systematic biases or gross errors due to measurement device failure. Data sets taken at a certain operation point are not model-consistent. Therefore measurement data have to be reconciled before used for model updating in order to improve the quality of the estimated model. The problem is that rigorous data reconciliation needs redundant data which are often not available. Therefore we performed data reconciliation based on simple component balances together with robust estimation techniques to identify gross errors and measurement biases. It can be seen that the quality of the updated model using reconciled measurement data is significantly better than using raw data.

We applied an online optimization with data reconciliation and model updating to the pilot plant for reactive desorption. The objective of the optimization was to reduce the heat duty needed for the desorption while simultaneously satisfying process restrictions and purity demands. In this contribution, we present the implementation of the online optimization with model updating and data reconciliation and show the results of the online optimization of the pilot plant.

The Process Model

Due to the chemical reaction in the liquid phase and the resulting enhancement in the absorption rate, absorption/desorption processes can not accurately be described with equilibrium models. Although more complex models are not automatically more accurate, in many cases short cut models are not adequate to reproduce the observed process. It has to be taken into account that a trade-off between model accuracy and computational complexity has to be made in selecting a model for simulation and optimization. This is especially important for online applications where it is required to solve the problem in real-time and biased results lead to wrong set points which makes frequent re-optimization runs necessary.

In this work the model proposed by Thielert [8] is used, because it has an adequate high accuracy and meanwhile an acceptable complexity. The model is a rigorous rate-based tray-by-tray model for the mass- and heat-transfer where the enhanced mass transfer due to the chemical reactions is accounted for by use of enhancement factors according to Hoogendorn et al. [5]. To describe the reactive system of weak and strong electrolytes in the liquid phase the following reactions are taken into account:

Primary reactions

$$H_2O \xleftrightarrow{K_1} OH^- + H^+ \quad (1)$$

$$NH_3 + H_2O \xleftrightarrow{K_2} NH_4^+ + OH^- \quad (2)$$

$$CO_2 + H_2O \xleftrightarrow{K_3} H^+ + HCO_3^- \quad (3)$$

$$H_2S \xleftrightarrow{K_4} H^+ + HS^- \quad (4)$$

$$HCN \xleftrightarrow{K_5} H^+ + CN^- \quad (5)$$

$$NaOH \xleftrightarrow{K_6} Na^+ + OH^- \quad (6)$$

Secondary reactions

$$HCO_3^- \xleftrightarrow{K_7} H^+ + CO_3^= \quad (7)$$

$$HS^- \xleftrightarrow{K_8} H^+ + S^= \quad (8)$$

$$NH_3 + HCO_3^- \xleftrightarrow{K_9} NH_2COO^- + H_2O \quad (9)$$

All reactions except for those including CO_2 and its products ((3), (7) and (9)) are considered as instantaneous reactions, since NH_3, HCN and H_2S are absorbed much more rapidly than CO_2.
To describe the chemical equilibrium relations containing the 15 species in the system, 15 equations are formulated:

9 reaction rate equations

5 component balances for the molecules included (NH_3, CO_2, H_2S, HCN and NaOH)

1 electron-neutrality relation

The mass transfer is described by a mass transfer coefficient k and the interfacial area $\bar{a}$. The correlations of the mass transfer coefficients and the interfacial area are taken from Billet and Schulte's [2] correlations.
Balance equations are used to describe the behavior of each stage in both the absorption and the desorption column including:

- *Stage component balances*
- *Vapor phase component balances*
- *Vapor-liquid equilibrium*
- *Liquid phase component balances*
- *Liquid-liquid equilibrium*
- *Summation equation for the vapor components*
- *Summation equation for the liquid components*
- *Heat transport equation*
- *Equation for the interfacial temperature*
- *Enthalpy balance*

The process model leads to a large-scale highly nonlinear equation system with 420 variables per ab-/desorption unit.

Parameter Estimation

In spite of the complexity of the model described above, there always exist mismatches between the model and real plant. A model validation through comparison of the simulation and experimental results shows a mismatch in the range of 20-30% for the industrial process [4]. Since model-based applications require a model that should be as accurately as possible, parameters in the model should be estimated to minimize the gap between the model and the real plant. To select the parameters to be estimated, we first analyzed their physical meaning and evaluated their impact on the process performance through simulation.
Following parameters were selected for the parameter estimation:

vapor flow parameter

$$C_{adj}^V = \Theta_1 \cdot C^V \quad (10)$$

liquid flow parameter

$$C_{adj}^L = \Theta_2 \cdot C^L \quad (11)$$

interfacial area

$$\bar{a}_{adj} = \Theta_3 \cdot \bar{a} \quad (12)$$

reaction velocity constant

$$k_{CO_2,adj} = 10^{\Theta_4 \cdot \left(13.635 - 2895/T_{B,s}^L\right)} \quad (13)$$

reaction velocity constant

$$k_{NH_2COO,adj} = 10^{\Theta_5 \cdot \left(11.130 - 2530/T_{B,s}^L\right)} \quad (14)$$

reaction velocity constant

$$k_{CO_3,adj} = 10^{\Theta_6 \cdot \left(13.635 - 2600/T_{B,s}^L\right)} \quad (15)$$

To improve the model accuracy, these process parameters were adjusted based on measurement data taken from the pilot plant. This leads to a large-scale nonlinear optimization problem that is to be solved with model-based parameter estimation techniques. As measurement errors have to be considered for dependent as well as independent variables the parameter estimation problem consists of two tasks: the parameter estimation problem itself and a data reconciliation step to rectify errors in the independent variables measurements. As the individual data sets are coupled over the parameters the two steps can not be solved separately. The size of the problem increases linearly with the number of data sets. In this study, 16 data sets were used for estimating 6 model parameters considering measurement errors for 15 independent variables for each data set. As a result, this problem includes more than 6500 optimization variables. Although commercial optimization software exist, which can handle large-scale nonlinear problems, highly nonlinear systems as the observed process are not easy to solve and often lead to convergence problems. Often extensive mathematical manipulations of the equation system or/and the optimization algorithm

are necessary to solve such a problem simultaneously.

We developed a new three-stage decomposition approach for parameter estimation [3] and applied this approach to solve this problem. The proposed approach uses a nested three-stage computation framework to decompose the problem (Figure 1).

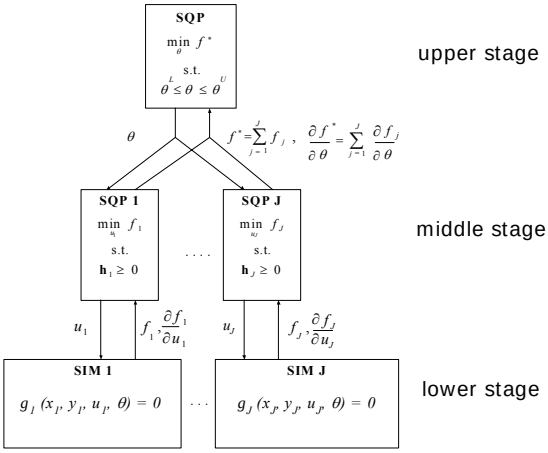


Figure 1: Three stage parameter estimation strategy

The upper stage is an NLP with only the parameters to be estimated as optimization variables. The middle stage consists of multiple sub-NLPs in which the independent variables of each individual data set are treated as optimization variables. In the lower stage the dependent variables are evaluated through a simulation step subject to the parameters and the independent variables. By making use of the optimality condition in the middle stage, only the gradients of the dependent variables to the parameters are required in the upper stage. The developed parameter estimation approach is able to handle large-scale systems with multiple sets of measurement data. Due to the reduction of the size of the optimization problem it is possible to use standard optimization software. The results after parameter estimation show a significant reduction of the errors between the model and the real plant. [4].

Data Reconciliation

The quality of the measurement data is crucial for the quality of the adjusted process model as well as for the quality of a model-based optimization itself. Usually measurement data contain stochastic errors, systematic biases or gross errors due to measurement device failure. The problem is that rigorous data reconciliation needs redundant data which are often not available. As no measurements for density data exist for the pilot plant (see next section), we perform data reconciliation based on simple component balances around the process considered (see the flow-sheet shown in Figure 2).

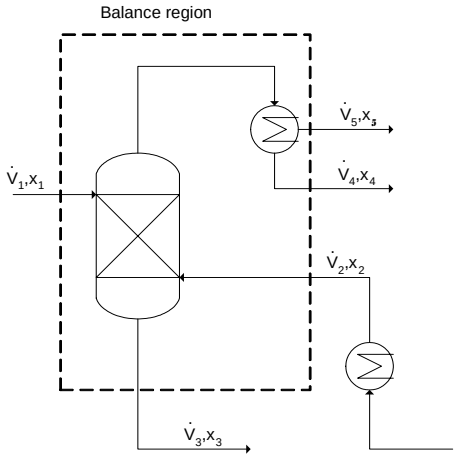


Figure 2: Simplified flowsheet to perform linear data reconciliation

For the pilot plant setup, the stream of uncondensed vapor (stream 5) can not be measured. As the experiment runs which were to be selected for parameter estimation were performed without reflux and therefore do not show high concentrations for NH_3, CO_2 and H_2S, the assumption that total condensation occurs can be made. As the solution contains sour as well as alkaline components, they dissociate completely in water and therefore the solubility is increased significantly. Therefore the assumption $\dot{V}_5 = 0$ is justified. Thus the component balances used for linear data reconciliation for desorption experiment runs are:

$$\begin{aligned} NH_3 &: \dot{V}_1 x_1^{NH_3} - \dot{V}_4 x_4^{NH_3} - \dot{V}_3 x_3^{NH_3} = r_{NH_3} = 0 \\ CO_2 &: \dot{V}_1 x_1^{CO_2} - \dot{V}_4 x_4^{CO_2} - \dot{V}_3 x_3^{CO_2} = r_{CO_2} = 0 \\ H_2S &: \dot{V}_1 x_1^{H_2S} - \dot{V}_4 x_4^{H_2S} - \dot{V}_3 x_3^{H_2S} = r_{H_2S} = 0 \end{aligned} \qquad (16)$$

The relative error for 16 observed data sets for the desorption experiment runs was in the range of 10-15%.

Generally the measurement errors do not exactly follow Gaussian distribution. Therefore, estimation of variables using least square objective function will give biased results. At the same time, due to the frequent presence of gross errors in the measurements, the reconciliation results are heavily contaminated. To overcome this obstacle, robust estimators could be used. A robust estimator is expected to be insensitive to large outliers, and

would have little effects on the reconciliation results when the error distribution is different from the assumed distribution.
Many robust estimators are available in the literature. In this study a robust estimator developed by Kong [6] was used. The estimator function is:

$$\min \quad \sum_{i=1}^{n} \frac{x_i^2}{\alpha + |x_i|} \qquad (17)$$
$$\text{s.t.} \quad f(\mathbf{x}) = \mathbf{0}$$

where $x_i = \frac{x_{m,i} - x_{r,i}}{\sigma_i}$ and α was set to 0.2. Standard deviations σ_i have been set to 5% of the measured value for concentration measurements and 7% for flow measurements. The reconciled measurement data was consistent with the component balances but showed deviations to the measured values up to 60%. Therefore measurement biases have to be considered. To identify these biases all 16 data sets were evaluated simultaneously. Suppose that system biases exist in the measurements of all measured variables, the conservation law of the process can then be described as f(x-bias,u)=0. Here the robust estimator described with the *Fair Function* by Albuquerque and Biegler [1] was used. Given the vector of measurements

$$\mathbf{x} = \left[\dot{V}_1, x_1^{NH_3}, x_1^{CO_2}, x_1^{H_2S}, \dot{V}_3, x_3^{NH_3}, x_3^{CO_2}, x_3^{H_2S}, \dot{V}_4, x_4^{NH_3}, x_4^{CO_2}, x_4^{H_2S}\right]$$

The following optimization problem was solved:

$$\min J = \sum_{j=1}^{16}\sum_{i=1}^{12} c^2 \left(\frac{abs\left(\frac{x_{j,i} - xm_{j,i}}{\sigma_{j,i}}\right)}{c} - \log\left(1 + \frac{abs\left(\frac{x_{j,i} - xm_{j,i}}{\sigma_{j,i}}\right)}{c}\right)\right)$$

$$s.t. \quad (x_{j,1} - b_1)\cdot(x_{j,2} - b_2) - (x_{j,5} - b_5)\cdot(x_{j,6} - b_6) - (x_{j,9} - b_9)\cdot(x_{j,10} - b_{10}) = 0 \quad , \quad j = 1,...,16$$
$$(x_{j,1} - b_1)\cdot(x_{j,3} - b_3) - (x_{j,5} - b_5)\cdot(x_{j,7} - b_7) - (x_{j,9} - b_9)\cdot(x_{j,11} - b_{11}) = 0 \quad , \quad j = 1,...,16$$
$$(x_{j,1} - b_1)\cdot(x_{j,4} - b_4) - (x_{j,5} - b_5)\cdot(x_{j,8} - b_8) - (x_{j,9} - b_9)\cdot(x_{j,12} - b_{12}) = 0 \quad , \quad j = 1,...,16$$
$$-0.5\min_j(x_{j,i}) \le b_i \le 0.5\min_j(x_{j,i}) \quad , \quad i = 1,...,12$$

An analysis of all 16 data sets showed that, before bias estimation, 16 relative residues indicate a deviation of >20%, and, after bias estimation, only 10. But 5 relative residues are deteriorated (4 for NH_3 and 1 for H_2S). Since the corresponding data sets showed no great deviation before bias estimation, they were excluded from the estimation problem. Performing bias estimation with the 12 remaining data sets, only 6 relative residues showed a deviation of more than 20% in contrast to 16 before bias estimation. The effectiveness of the bias estimation can be analyzed using the following criteria:

Apparent Number of Avoided Large Residues (ANALR)
Average of Residues After Estimating Biases (ARAEB)
Average of Residues Before Estimating Biases (ARBEB)
Improvement in Average of Residues (IAR): (ARBEB - ARAEB) / ARBEB

The analysis of these criteria is shown in Table 1. It shows a significant improvement for the deviations up to 65%.

Number of data sets		12
ANALR		10
NH3	ARAEB	0.115526
	ARBEB	0.339132
	IAR (%)	65.93
CO2	ARAEB	0.155545
	ARBEB	0.221069
	IAR	29.64
H2S	ARAEB	0.066982
	ARBEB	0.114219
	IAR	41.36

Table 1: Analysis of relative residues

The Pilot Plant

We use experimental data from a pilot plant established for both absorption and desorption studies on the industrial coke-oven-gas purification. The column is made of stainless steel with a diameter of 100mm and a height of 3m. It is packed with a structured packing of Sulzer Mellapak 350Y. Several measuring points along the column for temperature, pressure and concentrations are available. The plant is completely automated and all measurements and control variables can be collected and adjusted using the process control system Freelance 2000®. During operation several measurements for temperature, pressure, pressure drop and volume flow are available. Samples of the liquid phase can be collected automatically at four different column heights. However, the analysis of these samples to obtain measurement values for the liquid concentrations has do be done with chemical analysis and is therefore not available online.

Online Experiments

In order to perform online experiments including online model validation and online optimization, the simulation model and the optimization program were connected with the process control system to enable the exchange of measurement data and control variables between the process control system and the optimization program. Therefore a VBA (Visual Basic for Applications) interface was developed which organizes the data transfer together with the DDE Server (Figure 3). The DDE (Dynamik Data Exchange) is a standard process communication protocol for Windows® that allows the cyclic exchange of data with Windows® programs.

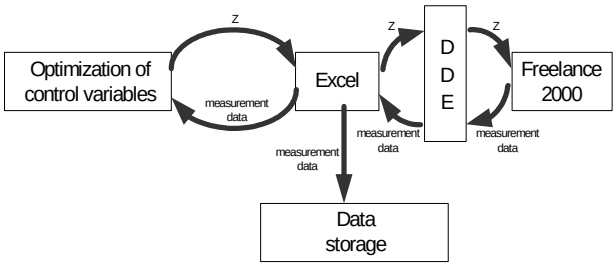


Figure 3: Schematic data interface

The described parameter estimation with data reconciliation was performed based on measurement data from various experiments from the past including absorption experiments. For the recorded data sets a significant improvement in model accuracy has been achieved. To use the estimated model in an online optimization structure, it has to be able to reproduce the actual state of the process and be able to predict the process behavior given certain control variables without further adjustment. Therefore we performed online model validation on the pilot plant, and simultaneously took the values of the input variables for simulation directly from the plant. In addition, set-point optimization was also carried out based on the measured data. The computed control variables were passed through the process control system to the plant. At the same time the measurements taken at the new steady-state point were compared with the prediction from the optimization result.

Set-point Optimization

The desorption operation with the pilot plant was considered. It was made according to the specifications of the deacidifier unit in the real scrubbing plant where the enriched water from the H_2S washing unit is freed from sour components. In addition the ammonia-rich water is siphoned off the unit to enhance H_2S absorption in the H_2S washing unit. To adjust the needed ammonia concentration, ammonia enriched steam is used for desorption. In addition the deacidifier was run in reflux mode. At the same time purity specifications at the bottom have to be satisfied. The feed specifications are given in Table.2.

	c NH_3 [g/l]	c CO_2 [g/l]	c H_2S [g/l]	c NaOH [g/l]	$\dot{V}$	T [°C]
vapor feed	4,5	0	0	0	15 kg/h	-
liquid feed	10	7,2	2	2	118 l/h	80

Table2: Feed specifications used in online experiments

A steady-state operating point was first established and the measurement data were collected online using the interface described. The first steady-state set-point is shown in Table 3. Based on the measured data an optimization computation was made. The objective was to minimize the heat duty needed for the desorption operation. Three optimization variables were chosen:

- Heat duty $\dot{M}_{steam}$
- Liquid feed temperature T_{liq}^{feed}
- Reflux ratio R

According to the real process operation, following restrictions were included:

- NH_3 concentration at h=856mm $\geq$ 20 g/l
- H_2S concentration at bottom $\leq$ 0.5 g/l

Thus the following optimization problem had to be solved:

$$\begin{aligned} &\min_{\dot{M},T,R} f = \dot{M}_{steam} \\ &\text{s.t.} \\ &h_1 = c_{NH_3}^{856} - 20 \geq 0 \\ &h_2 = c_{H_2S}^{bottom} + 0.5 \geq 0 \end{aligned} \tag{18}$$

A sequential optimization approach was used to solve this optimization problem, where only the objective function and the inequality constraints are handled by the optimization algorithm (a standard SQP routine from the IMSL FORTRAN library). The state variables are calculated in a

simulation layer which provides the objective function and the gradient information.

	$T_{liq}^{feed\,\dagger}$	$R^{\dagger}$	$\dot{M}_{steam}$ †‡
First set point	80 °C	0	15 kg/h

Table3: Control variables used for first stationary set point

Results

At first, the quality of the fitted model was checked with the data sets used for offline parameter estimation. Figure 4 shows the liquid concentration profiles for one data set used in parameter estimation for NH3, CO2 and H2S. It can be seen that the quality of the model was significantly improved with the estimated parameters (solid lines). Considering all 16 data sets the mean model deviations could be reduced from 30.70% to 14.77% in the desorption case and from 28.09% to 15.36% in the absorption case.

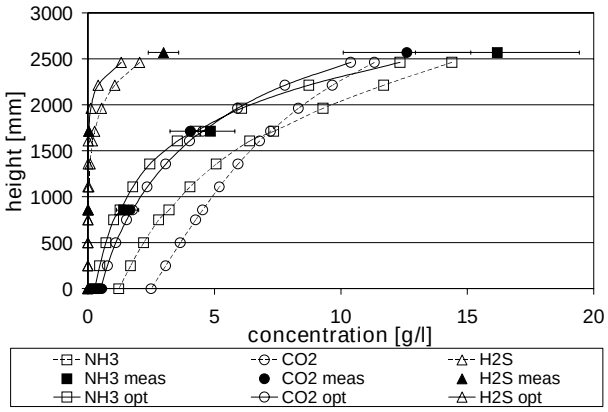


Figure 4: Liquid concentration profiles for one data set with and without parameter estimation

To validate the model online, an online optimization experiment described in the previous section was carried out. At the steady-state shown in Table 3, the measured values for the liquid concentrations are compared with the result from the simulation based on the measurement values in Figure 5. It can be seen that a good consistency between measured and simulated data can be received. Only for the H_2S concentrations a deviation of more than 20% can be observed. This may be caused by measurement inaccuracies for small concentrations or model deficiencies which have been observed for H_2S mass transfer prediction. This problem was noticed and a bypass model has been developed to account for maldistribution effects inside the column [7]. Neglecting the H_2S concentrations the mean model deviations are 10.4%.

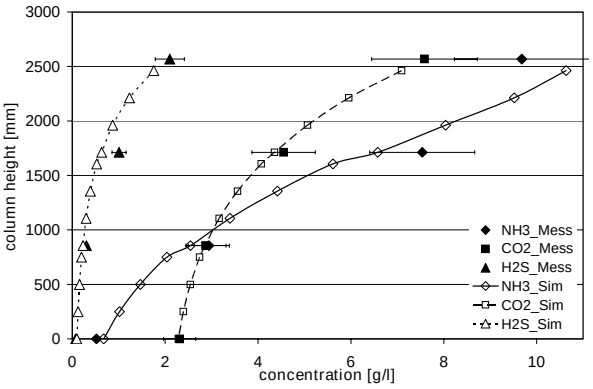


Figure 5: Liquid concentration profiles at first stationary point

After the first steady-state had been established, the measurement data from the plant were used to perform the online optimization. The evaluated values for the optimization variables are given in Table 4. According to the optimized results, the liquid feed temperature should be increased to its upper bound to increase the mass transfer and to reduce condensation in the upper part of the column. This helps meeting the process restriction at the bottom of the column. The heat duty was reduced to minimize the operation cost without violating the process restrictions for the concentration of NH_3 at h=856mm. In contrast, the reflux ratio was increased in order to increase the concentration of NH_3 without violating the purity specification at the bottom.

	$T_{liq}^{feed\,\dagger}$	$R^{\dagger}$	$\dot{M}_{steam}$ †‡
First set point	96°C	3,8	10,32 kg/h

Table 4: Control variables after process optimization

The computed new control values were passed through the process control system to the plant. Thus the new steady-state was established. The liquid concentration measurements are compared with the results from the optimization in Figure 6. It can be seen that a good consistency between the measured values and the optimization exists. Especially the process restrictions were met quite well with 3.7% deviation for the restriction on NH_3

† manipulated variable
‡ objective value

at h=856mm and 6.8% deviation for the restriction on H_2S at the bottom.

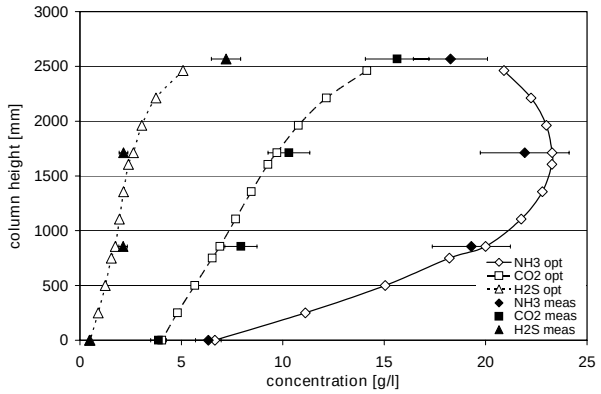


Figure 6: Liquid concentration profiles after optimization

Conclusions

A rigorous rate-based process model was used to describe the mass transfer for the *ammonia hydrogen sulfide circulation scrubbing* process. Several model parameters were selected and adjusted with a three-stage sequential parameter estimation routine based on 16 measurement data sets. The parameter estimation approach is able to handle large-scale nonlinear process models with an arbitrary number of data sets. Due to the reduction of the degree of freedom standard optimization software can be used. Data reconciliation based on simple component balances was performed to obtain model consistent measurement data and to identify measurement biases. With the parameter estimation based on the reconciled data, the average model deviations could be reduced significantly. For online model validation the process control system was connected with the simulation program via a user friendly interface to exchange measurement data and control variables. To test the online applicability of the model, process optimization was carried out with a sequential optimization approach based on measurement data taken online from the plant. The objective was to reduce operation cost by minimizing the reboiler heat duty. Process restrictions were included according to the real scrubbing process. The results show that a good agreement between the measured values and the results of the optimization can be achieved and that the process model can be used to build a complete online optimization structure for the *ammonia hydrogen sulfide circulation scrubbing* process.

Acknowledgement

Financial support to this work from the Deutsche Forschungsgemeinschaft (DFG) under the project contract LI806/4-1 is gratefully acknowledged.

References

[1] Albuquerque, Joao S.; Biegler, Lorenz T. *Data reconciliation and gross-error detection for dynamic systems*. AIChE Journal 1996; 42(10): 2841-2856

[2] Billet, R.; Schultes, M. *Predicting mass transfer in packed columns.* Chem. Eng. Technol. 1993; 16(1): 1.

[3] Faber, R.; Li, P.; Wozny, G. *Sequential parameter estimation for large-scale systems with multiple data sets. Part I: Computational Framework.* Ind. Eng. Chem. Res. 2003; 42(23): 5850 - 5860.

[4] Faber, R.; Li, P.; Wozny, G. *Sequential parameter estimation for large-scale systems with multiple data sets. Part II: Application to an Industrial Coke-Oven-Gas Purification Process.* Ind. Eng. Chem. Res. (accepted for publication 2004).

[5] Hoogendorn, G. C., Abellon, R. D., Essens, P. J. M., Wesselingh, J. A. *Desorption of volatile electrolytes in a tray column (sour water stripping)*. Chem. Eng. Res. Des. 1988; 66: 483-502.

[6] Li, Bo; Chen, Bingzhen; Duan, Dongyong; Kong, Mingfang. *Correction coefficient method for gross error detection based on temporal redundancy*. Qinghua Daxue Xuebao, Ziran Kexueban 2000; 40(10): 65-69.

[7] Thiele, R.; Wozny, G. *Consideration of maldistribution in an online model for a gas treatment control system for chemical absorption and desorption in packed columns.* In: ECCE, Granada, 2003.

[8] Thielert, H. *Simulation und Optimierung der Kokereigasreinigung.* Dissertation, Institut für Prozess- und Anlagentechnik, TU Berlin, 1997.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

MODEL BASED CONTROL AND OPTIMIZATION IN PULP INDUSTRY

Johan Jansson *) **) and Erik Dahlquist *)
*) Malardalen University, Box 883, SE 721 23 Vasteras , Sweden
**) ABB Automation, SE- 721 67 Vasteras , Sweden

ABSTRACT

In this paper we present a combination of the usage of a model for the whole mill on-line simulation and more advanced models for specific process sections for control systems and online diagnostics. Different solution methods are described together with where the different approaches are best suited.

BACKGROUND

The trend today is to optimize the whole mill not only towards production and quality, but also towards for example minimization of energy and chemical consumption, and effluents. Most of the models are first principles process model, which is physical models. These are used to simulate the process as part of the optimization. With a good process model it is possible to implement a more advanced control system like model predicted control (MPC) and 2D or 3D dynamic online models for diagnostic purpose. E.g. hang ups and channeling are of interest to identify in a continuous digester.

For the on-line simulation of the whole mill ABB has developed a Pulp and Paper On-line Production Optimizer and this optimizer works like an "overall mill management system" giving set points to each process section. The models will be on the same software platform as the more advanced models, for example the digester model presented in this paper.

PROCESS OPTIMIZATION AND CONTROL

In this paper we will focus on the use of adaptive physical models for both on-line control and production planning the next 24 hours. In reality the boarders between statistical models and physical models may not be that clear. If we introduce a number of parameters into a physical model and these have to be tuned by plant data, it is in reality a combined physical and statistical model. The advantage is that we get the robustness of the physical model, but can make use of the statistics really relating to the actual process.

On the overall mill level we want to optimize and control the chemical balance in a complex pulp mill. The first application is the control of Sulfur in an integrated Kraft and NSSC (Neutral Sulfite) pulp mill. It is important to adjust the dosage of sulfur to adjust the sulfidity, so that there is enough but not too much, and to keep the balance between different parts of the plant. This has been done by simplified physical models that have been combined to form a model. This model is then used to make a production plan for keeping the sulfur balance at it's optimum. Implementation is being done right now at a mill in Sweden (Billerud at Grums).

The next level of optimization is to look inside each process section, in our case a continuous digester. For digester operations we have been working with physical modeling of the digester in both 1-D and 2-D.

The physical model of the digester is built on the same principle as the so called Purdue model (Wisnewski et al.1997). The model contains two volumes, the volume occupied by the chips with the entrapped liquor, and the volume occupied by the free liquor. The model also contains similar chemical reaction and energy balances as the Purdue

model. What is new with our model is that we also taking into consideration the pressure drop inside the digester due to the channels between the wood chips. When there are mostly large chips, the channels between the chips are large, with low pressure drop for fluid flowing between. When there are a major amount of fine pins etc, the channels between the chips will decrease, and the pressure drop increase. This is what happens in reality if we have different chip size distribution or different packing in the digester. One reason for this may be that the chip size distribution is inhomogeneous or the chip screw in the stack may not feed in a constant way. When we have a lot of flakes these may adhere to the screens and cause hang ups. Aside of causing an increased pressure drop in the screens, also the chips will get different residence times and contact with liquors of different concentration of both chemicals and dissolved organics. This may cause a significant variation in the kappa number of the final fibers. The kappa number is a direct measure of the remaining lignin at the fiber surface. By identifying pressure drops, residual concentration of chemicals in the liquors, temperatures and flows and compare actual results to those predicted by the model, we can tune the model to match reality. This is under the assumption that we first have achieved a good process performance of the actual digester. The model can then be used both to optimize the performance by adjusting e.g. temperature and chemicals dosage, as well as back flushing screens to avoid hang ups before the problems become severe. There is also a potential for finding channeling to have a chance to go in and adjust, although this demands regularly measurements as well.

The modeling can be used in several different ways. The first approach was to principally use a sequential approach, where the fibers and water is supposed to flow from the top to the bottom, with chemical reactions taken place in each volume element each time step of calculation. The pressure flow calculations then are calculated as well for each time step for a identified pressure flow network, with given boundary conditions. There is then iteration between the pressure flow calculation for the whole network and what is happening in each volume element at each time step. The advantage with this approach is that we always get stable solutions, and the calculation time is more or less the same each time step, independent of dynamic changes. The draw back is that the accuracy may be a bit less good than if the other approach is used, that is using a simultaneous solver.

In this approach with a simultaneous solution of both chemical reaction calculations and pressure flow network calculations, we principally get a better solution, and normally the time step is depending on if a step change in the process is taking place or not. When there is a fast event taking place, the time step becomes very short, to get accurate solutions. The good thing with this is obvious, but the draw back is that you have no idea in advance how long the calculation will take. When running a simulation in parallel with a real process this may give big problems, as the calculation time can be many times longer than “normal”, and the real process “passes” the simulator!

DIFFERENT MODELING APPROACHES

We have tested a number of different approaches for the simulation of the digester. They are presented shortly in the following:

1. **Sequential solver** using Fortran code, with an iteration between pressure-flow calculations and chemical reactions-tank level calculations. 1-D.

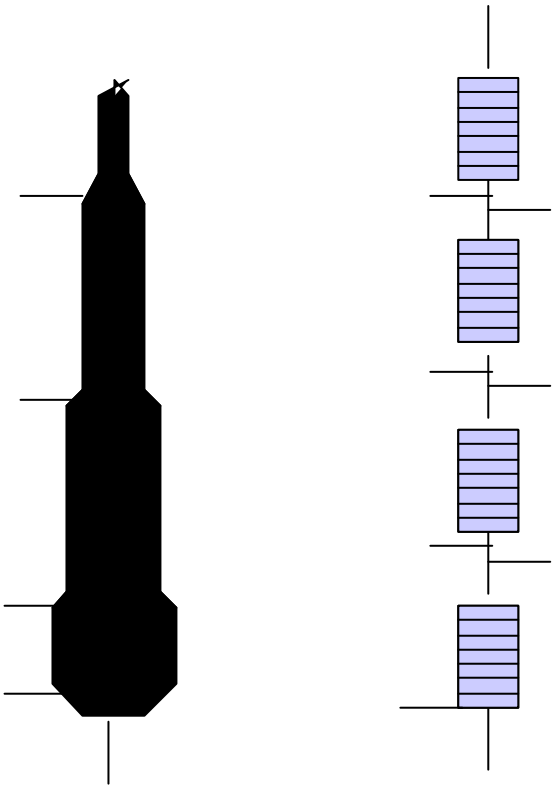


Figure 1. Use of a sequential solver

First the calculations of pressure is performed by looking at pressure lift by pumps, levels and pressure head in tanks, pressure drops in valves and entrainments. This gives the actual flows in each single pressure point, as we know the characteristics of the pumps, valves, valves openings etc.

In the second step the flow rates are used to calculate flows into a volume element and out of it, giving new levels in the tanks/vessels.

In the third step we use the actual temperatures and concentrations in each volume element to calculate chemical reactions, diffusion, transfer between phases etc. These are then used to calculate the new concentrations of all the flows in and out, as well as the new concentrations in the different phases in each volume element. In the figure to the right we have real or "faked" valve between all the arrows, giving the pressure at all points separating the arrows. Real valve data are used for the dimensions, but tuning has to be done to give correlations between the actual flows as a function of chip size distribution, viscosity of the mixtures, temperature etc.

This approach is well suited for operator training of a system, and to be combined with a DCS system, with full controls.

A variant of this approach was used by Bhartia and Doyle (2004) to model plugging in the digester, and laso other types of engineering use is common, to study specific design or operational problems.

2. **Simultaneous solver but without pressure flow calculations**. The flows are assumed controlled by the real DCS system. 1-D

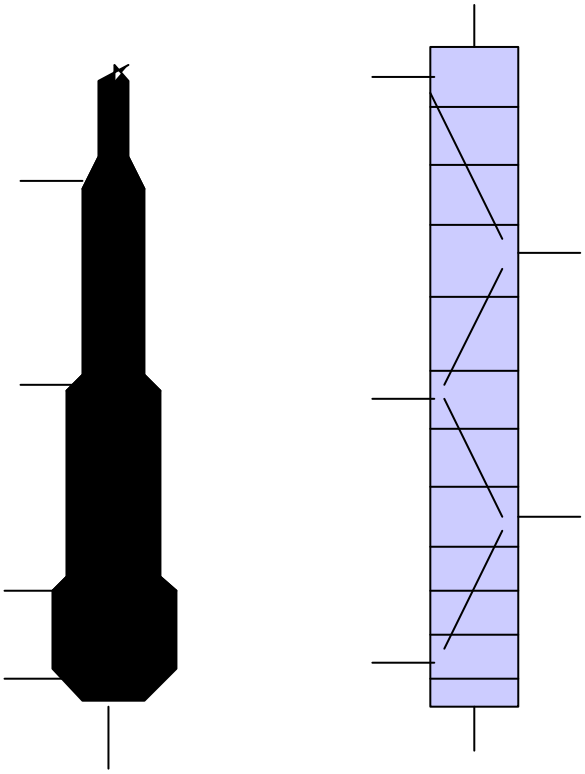


Figure 2. Use of a simultaneous solver but with fixed flow directions

In this case we assume all flows being controlled by the DCS system, and the direction of the flows are always fixed- up or down. This makes it easier to calculate the concentrations, as we know what concentration is going what direction. If you have alternating flow directions as the pressure drop is changing in a section, it takes more calculation power to find a stable solution with both correct concentrations and correct flows, otherwise. In this approach with simultaneous concentration and flow calculations we get more correct concentrations, specially during transients, but more unpredictable calculation times. The advantage with not including pressure flow calculations is that it is easier to use this model for MPC-calculations, where you want to find an optimal setting of the different variables like temperature, concentrations and flows, than if you included also pressure flow calculations.

3. 2-D calculations with a simultaneous solver for both pressure flow calculations and chemical reactions

In this approach with 2-D we have a "fake valve" between each volume element, giving a complex net inside the digester, but also including real valves in and out of the digester. With this approach we calculate the flow between the volume elements, and can simulate hang ups, channeling, different packing etc. The model is used together with measurements of real flows, chip size distributions, chemical additions, concentrations in black liquor in extractions, temperatures, final kappa number of the chips etc.

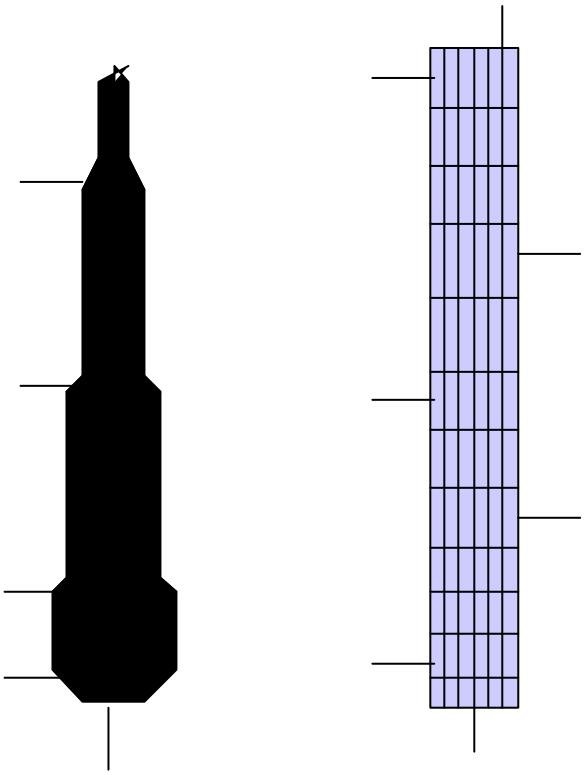


Figure 3. Use of a 2-D model with a full pressure-flow network

This model can easily be extended to 3-D, although the computation time goes up significantly with 3-D compared to 2-D.

This gives the opportunity to use the model for different type of diagnostics as well as for testing of different design and operational options.

The digester model will also be used for sensor validation for the measuring apparatus around the digester. If the calculated value will start to differ to much from the measured value, that will indicate in an early stage that there are some problems with that measure point. An early indication that there are some problems will save production, quality and money. This type of sensor validation is important as a pre-step to the actual optimization.

The advanced physical model can among others be used to control the process in a MPC (model predicted controller). MPC is a multivariable model predictive control technology. The main advantage with MPC is its ability to handle multivariable processes with strong interaction between process variables and with constraints involving both process and state variables. The potential of this control method is dependent on the quality of the process model and of the measured variables. With a good mass balance simulation of the whole mill the MPC is less dependent of the measured variables. All three models described above can be used, although the number two model is the one we have been testing as the first choice. As the MPC is giving set points to the different control loops, the pressure flow net calculations are normally of less interest.

So far we have tested this in a simulation environment with good results. Next step is to make real application tests in a mill.

When it comes to production planning an scheduling, including “tank-farming”, we

have to model not only the digester, but also all other equipment surrounding this. There are several approaches for the modeling here. We can use the approach with models for tanks and some other features directly in the formulation of the Objective function and the constraints, and solve the dynamic optimization problem directly. Another alternative is to use a dynamic simulation model, and use this to calculate the result for the whole time horizon using a "branch and bound" type of solver. Then communication is intense between the optimizer and the simulator, but on the other hand we know we get a feasible solution. A third approach is to use the optimization using the first alternative, and then test this solution towards the more detailed simulator, to avoid running into possible problems implementing the optimal schedule.

All these approaches has been implemented, but for different applications. In the EU DOTS project (Bell et al ,2004 and Dhak et al 2004), we tested all these approaches, but for paper mill applications. At Gruvon pulp mill, the approach was the first alternative. Here the problem was very large and complex, where a large number of tanks and processes had to be included. To avoid problems with unreliable sensor measurements the signals were filtered through a "moving window" approach, which turned out to be quite successful. This was done by ABB (Persson et al 2003).

RESULTS

The **first approach** gives dynamics for the complete mill, and interactions between all equipments. It is very useful as a **training simulator** and teaches you how the DCS functions like interlockings, PID controls etc respond. An example of a response to a change in a flow rate is shown in the figure below. Here we get the process values directly at a process display, just like in the real plant. Trends can be shown of variables, PID control response etc.

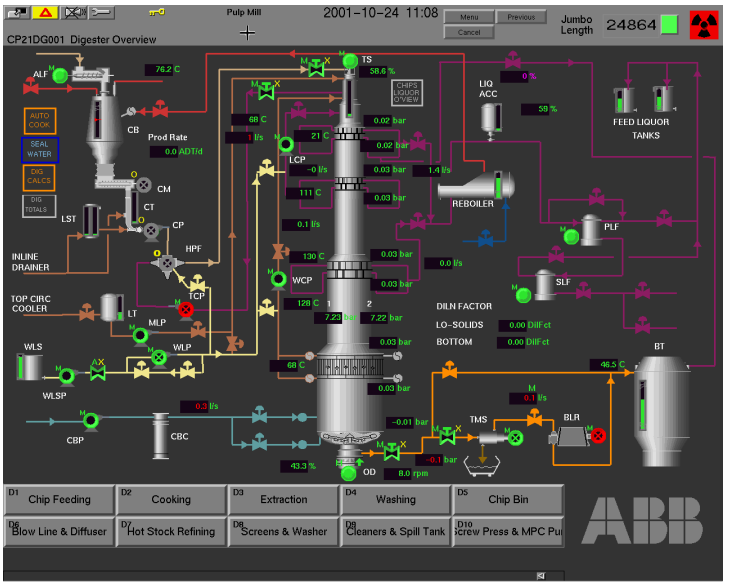


Figure 4. A process display from a training simulator, where a DCS system is interacting with a dynamic process simulator on-line.

The **second approach** with a simultaneous solver for all digester equations is suitable to use for **MPC** (Model Predictive control) control. An example of how the optimized values for all important control variables looks like compared to the "normal recipe" is shown in the table 1 below. **The third approach with a 2-D model** is best to use for **diagnostics** purpose as to determine channeling in the digester, or hang ups. Channeling will mean that liquid is not getting into the chips, but passing in cahnnels between the chips. This means that we will get less reaction between chemicals and lignin, resulting in less dissolved lignin (DL) in the extraction liquor, as well as more residual alkali (NaOH), as not all got the chance to react. An example of how this can look is seen in the simulation in figure 5.

Table 1 Optimized values using an MPC compared to what values should have been used with the normal practice recipe.

		Base	Optimized	Description
Decision Variables:				
	SI103A	22,421	22,421	Chip Screw Rate, rpm
	FI107A	2,499	2,370	White Liquor to High Pressure Feeder, lit/s
	FI104B	4,463	4,240	White Liquor to Digester, lit/s
	FI217B	2,701	2,836	White Liquor to Wash Circulation, lit/s
	FI216B	6,578	6,907	White Liquor to Lower Cook Heater Circulation, lit/s
	TI216A	147,387	136,874	Lower Cook Heater Outlet Temperature, degC
	TI217A	153,017	160,274	Wash Circulation Heater Outlet Temperature, degC
	TC102A	94,000	99,000	Chip Bin Temperature, degC
	FI216C	6,467	6,790	Wash Liquor to Lower Cook Heater Circulation, lit/s
	FI217C	6,363	6,048	Wash Liquor to Wash Circulation, lit/s
	FI212H	42,237	42,238	Dilute Filtrate, lit/s
	FI104C	0,108	0,102	Wash Liquor to Chip Tube, lit/s
	FI102A	1,220	1,354	Reboiler Steam, kg/s
Dependent Variables:				
	Pulp Produced	6,56	6,62	kg/s
	Chip Utilization	26,067	25,976	kg/s
	Dissolved Lignin	5,680	5,495	kg/s
	NaOH Consumption	0,713	0,698	kg/s
	Na2S Consumption	0,406	0,405	kg/s
	LP Steam Utilization	(0,069)	0,001	kg/s
	MP Steam Utilization	1,653	1,499	kg/s
	Steam Condensate	(0,130)	0,003	kg/s (negative value indicates net production)
	Total White Liquor Utilization	16,241	16,353	lit/s
	Total Wash Liquor Utilization	55,175	55,178	lit/s
	Black Liquor	51,521	51,545	lit/s
	Kappa Number	89,56	89,56	
	Pulp Yield	57,45	58,22	%
	Objective Function	0,047839	0,074172	US$/s

In figure 6 below we can see what this means for the temperature profile from the top to the bottom of the reactor. The dark (blue) line is how the set-points would aim at if we were operating according to the normal strategy for this type of wood. The brighter (red) line shows how the MPC proposes to operate, to get lower energy and chemical consumption, but with the same kappa number (remaining surface lignin) out of the reactor.

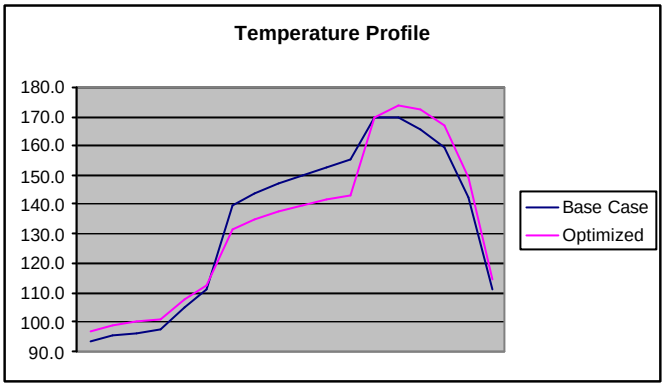


Figure 6. Temperature profile in the digester from top to bottom according to "normal operation" (dark blue) respectively as proposed by the MPC (brighter red)

CONCLUSIONS

In the presentation we have discussed different means for modeling digesters to use for MPC and optimized scheduling. The first approach with a dynamic simulation model with iteration between pressure-flow net calculations and reactions in each volume element is suitable for dynamic simulation, were also control actions through the DCS system has to be encountered. This method is suitable for detection of different faults, testing of "what if scenarios" and can be used in optimization interactively with an optimization algorithm. The second approach without a pressure flow net work solver, but with a simultaneous solver for calculation of all reactions taking place inside the digester is well suited to use for MPC applications, where set points are to be given to the control loops. The third approach with more detailed models in 2-D (or even 3-D) is best suitable for detection of hang ups, channeling and other type of faults in the process. In the future, when computer capacity is significantly higher, it should also be possible to use this instead of the two other types of models.

REFERENCES

Bell J.,Dahlquist E.,Holmstrom K.,Ihalainen H.,Ritala R.,Ruis J.,Sujärvi M.,Tienari M: Operations decision support based on dynamic simulation and optimization. PulPaper2004 conference in Helsinki, 1-3 June 2004.Proceedings.

Bhartiya S and Doyle F J (III): Mathematical model predictions of plugging phenomena in an industrial single vessel pulp digester, will be published in Ind&Eng Chem Res during 2004.

Dhak J.,Dahlquist E.,Holmstrom K.,Ruiz J.,Bell J.,Goedsch F: Developing a generic method for paper mill optimization.Control

Systems 2004 in Quebec City, 14-17 June 2004. Proceedings.
Ulf Persson, Lars Ledung, Tomas Lindberg, Jens Pettersson, Per-Olof Sahlin and Åke Lindberg: "On-line Optimization of Pulp & Paper Production", in proceedings from TAPPI conference in Atlanta, 2003.

Wisnewski P.A, Doyle F.J and Kayihan F.: Fundamental continuous pulp digester model for simulation and control. AIChE Journal Vol 43, no 12, dec 1997, pp 3175-3192.

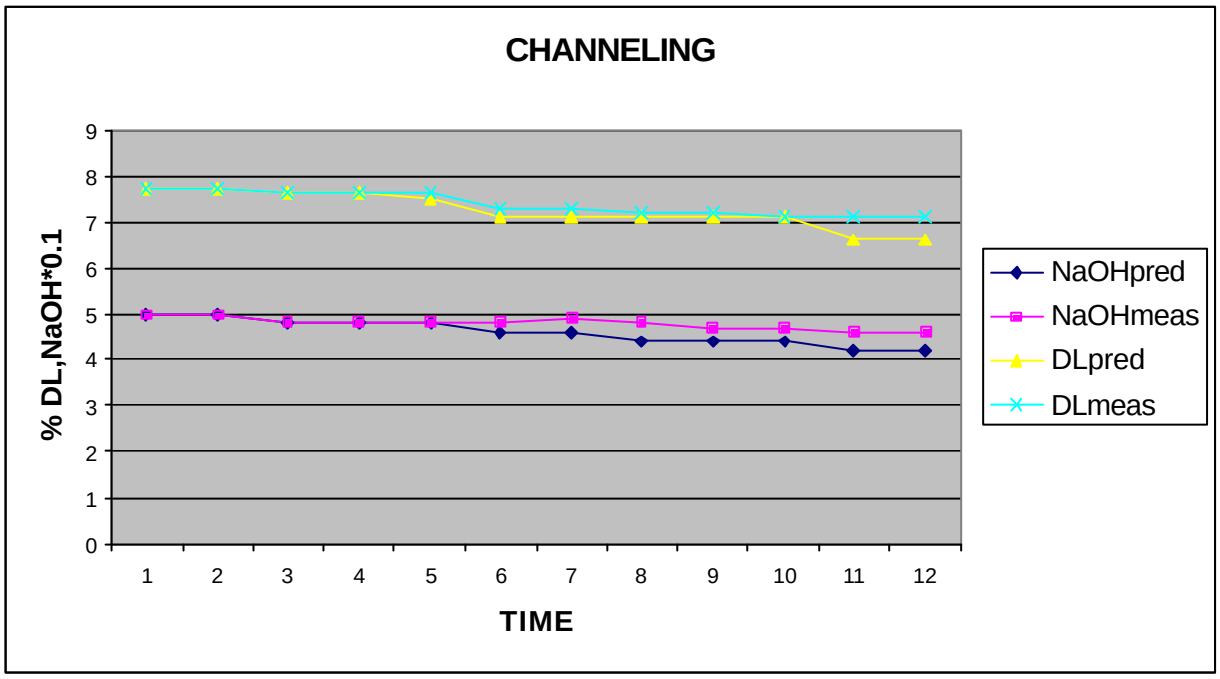


Figure 5 Simulated values for concentrations of NaOH (‰) respectively dissolved lignin (% DL) in the extraction line liquor. Predicted values are from normal operations and measured values correspond to values when there is channeling.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

METHODS FOR HANDLING UNCERTAINTIES IN DYNAMIC SIMULATION AND OPTIMIZATION: CASE TMP PLANT

Petteri Pulkkinen[1], Risto Ritala

Institute of Measurement and Information Technology,
Tampere University of Technology, P.O. Box 692, FIN-33101 Tampere

Abstract

The optimization problems for industrial processes have three characteristic properties which make them difficult to solve. Firstly, the decision variables often have to be integer, for example: the number of refiners running. Secondly, most of the measurements are stochastic. Ignoring the effect of variability, for instance in process times, would largely limit the practical use of the outcome of a simulation. Thirdly, the analysis of the problem can be computationally expensive, especially for more complicated models. Seeking help in designing complex industrial systems, we need a design optimization strategy that can deal with all three of these characteristics. In this paper we take steps to include stochastic elements to the simulation and optimization.

KEY WORDS: stochastic, simulation, optimization

Nomenclature

$< F(t)>$	average (of white noise)
f_0	deterministic flow
f_{out}	outflow
f_{unit}	production speed of one refiner
P	probability
X_n	random variable
$N(t)$	refiner schedule
M	sequence of random variables
$V(t)$	volume of the tank
$F(t)$	white noise
Ω	discrete state space
σ	standard deviation
τ	t-t_0

Introduction

The process to be optimized is the production of thermo-mechanical pulp (TMP). TMP is produced by steaming wood chips under pressure to break up the wood structures followed by mechanical refining to produce pulp. TMP plants use electrically operated refiners; the demand for the energy is huge. Electricity consumption comprises a large amount of the variable production costs and offers an area of optimization and cost savings.

The optimization case is actually a continuous decision support for running a plant of identical on/off processes under time-variant production costs, time-variant needs of the product and limited intermediate storage capacity. The objective function, required by the optimization, is evaluated through running a dynamic simulation. The simulation model is based on the TMP pulp production and flows, predictions of the TMP demand and time variant electricity costs. The objective of the deterministic case is to minimize overall electricity costs while taking into account the limited tank volumes between TMP plants and paper machines.

[1] Corresponding author. Phone: +358 40 564 3184,
FAX: +358 40 3115 2171, E-mail: petteri.pulkkinen@tut.fi

As the simulation requires predictions about the future, we must decide how to deal with the uncertainty. It is possible to proceed to optimize the TMP production assuming the predictions to be accurate. This assumption is well motivated if

a) the probability distribution of predicted properties is narrow around mean
b) the width of the distribution is similar for all TMP production scenarios

When the assumptions are not well motivated the optimization must deal with other aspects of predictions than predicted mean.

In this paper we discuss how the uncertainty enters to the objective through e.g. risk premium. Then we present methods for handling uncertainties in dynamic simulation and optimization. Instead of mean or fixed (guessed) values, the full probability distributions are used. Stochastic factors in the model are either white noise type of uncertainty caused by the process fluctuation, or the probability for the line break in the paper machine. Added stochasticity makes the simulation and optimization results more realistic. The optimization tool actually supports the decision making by giving not only the hard results but also the figures for evaluating the probability of the good decision. This leads to the fact that the process is optimized also psychologically, through improved user acceptance.

Description of the case

Paper consists of a mix of fibers, mineral fillers and chemicals. In particular, in printing paper mechanically disintegrated fibers are the main ingredient. Thermo-mechanical pulping is one of the two most common mechanical fiber disintegration technologies.

TMP is produced by disintegrating fibers through a mechanical action on wood chips in a high temperature and pressure environment generated by steam addition. TMP plants consist of electrically operated refiners. TMP plant contains typically of 3-8 parallel production units, each having in series a primary refiner for disintegrating the fibers and a secondary refiner to trim their properties for the end product. In what follows we shall consider the pair of primary and secondary refiners as a single production unit and call it a “refiner line”, or where obvious, simply a refiner. Furthermore, due to quality and cost reasons, it is sensible to operate a refiner only at full production capacity and therefore a refiner’s state can be referred to as a binary variable, or as either “on” or “off”.

To ensure the quality properties of TMP, the pulp is screened and further improved with reject refiners. We shall assume the operation of screening and reject refiners as constant but with capacity limitations. The production of refiner lines is collected to one or several intermediate tanks, out of which they are distributed to one or several paper machines.

The process requires a large amount of energy, 2.0-3.5 MWh/ton pulp and a TMP plant produces between 500-1500 tons a day. Much of the energy is transferred into steam and is recovered for the drying area of the paper machines. Electricity consumption comprises a large amount of the variable production costs and offers an area of optimization and cost savings, since the electricity cost varies strongly during one day.

The plants are designed to produce enough pulp while running below full capacity to allow for plate changing, process fluctuations, unscheduled maintenance and other operational constraints. This overcapacity opens up possibilities to optimize electricity costs based on the margin provided by daily fluctuations of the free market electricity price. A TMP production schedule must meet the demands of the paper machines while not exceeding the storage capacity between the TMP plants and PMs. The storage capacity is large enough to allow electricity cost optimization of schedules.

In order to minimize the operating costs, the TMP production schedule will be optimized around the market energy price and constraints of the mill. The model to predict TMP plant behaviour is extremely simple: the plant consists of maximum 5 identical refiners each producing TMP at a known rate when on. The flow from each refiner goes to a storage tank from which a time variant demand of TMP is taken. The demand is predicted on the basis of production schedule. The prediction of electricity cost is available, and the time-invariant costs of refiner startup and shutdown are known. Figure 1 illustrates the model.

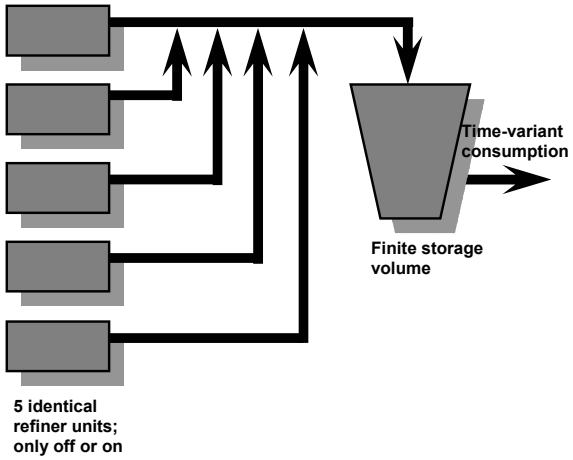


Figure 1. A simple model for TMP plant, to be used in refiner scheduling optimization.

The main information of the example case is listed here:

- Optimization target is to minimize the electricity cost
- Variables which can be manipulated are startups / shut downs of the refiner lines
  - Changes are step changes
- Optimization horizon is 48 hours
  - TMP demand (based on paper machine's production schedule) for the optimization horizon is known
  - Electricity price for the optimization horizon is known
  - Decision interval is 15 minutes
- The number of running refiner lines can vary between 0 and 5
- Maximum number of the changes during the simulation horizon is 5
- Power of each refiner line is 11 MW
- TMP production speed is 3.5 t/h/refiner
- The volume of the intermediate tank is 500 m^3

Stochastic elements of the case

Essentially all production decision support systems neglect the stochasticity. This is the most limiting factor of their applicability. In this paper stochasticity is present in our simplified TMP production model in two ways: as a white noise type process fluctuation and as uncertainty caused by the random breaks of the paper machine.

White noise type fluctuation

The volume of the modelled intermediate tank is controlled by the incoming and outgoing flows. Incflow is defined by the refiner schedule and the production speed of the refiners. The outgoing flow is defined by the demand of the paper machine that is predicted on the basis of production schedule. The outflow f_{out} is a sum of deterministic part and white noise and can be expressed as

$$f_{out}(t) = f_0(t) + F(t). \tag{1}$$

The properties of white noise are defined with following equations:

$$\begin{aligned} &\langle F(t)\rangle = 0 \\ &\langle F(t)F(t')\rangle = \sigma^2\delta(t-t') \end{aligned} \tag{2}$$

The volume of the intermediate tank as a function of time can be expressed with

$$\frac{dV}{dt} = N(t)\cdot f_{unit} - f_{out}(t), \tag{3}$$

where $N(t)$ is the refiner schedule and f_{unit} is the production speed of one refiner.

Starting from equation (3) and using equations (1) and (2), an equation for the probability distribution of the volume of the tank can be analytically derived. The equation is defined as

$$\mathrm{P}\left(\int_{t_0}^{t_0+\tau}\left(N(t)\cdot f_{unit} - f_{out}(t)\right)dt + V(t_0) = v\right) = \left(2\pi\sigma^2\tau\right)^{-1/2} exp\left[-\frac{\left(v - V(t_0) - \int_{t_0}^{t_0+\tau}\left(N(t)\cdot f_{unit} - f_{out}(t)\right)dt\right)^2}{2\sigma^2\tau}\right], \tag{4}$$

where σ is the standard deviation and τ is $(t\text{-}t_0)$.

In order to illustrate the meaning of previous equations we can select an optimization scenario and have a look at the graphs. Figure 2 presents three time series needed by the optimization. Inflow is defined by the refiner schedule, deterministic part of the outflow is predefined and the expected value of the volume is calculated based on flows.

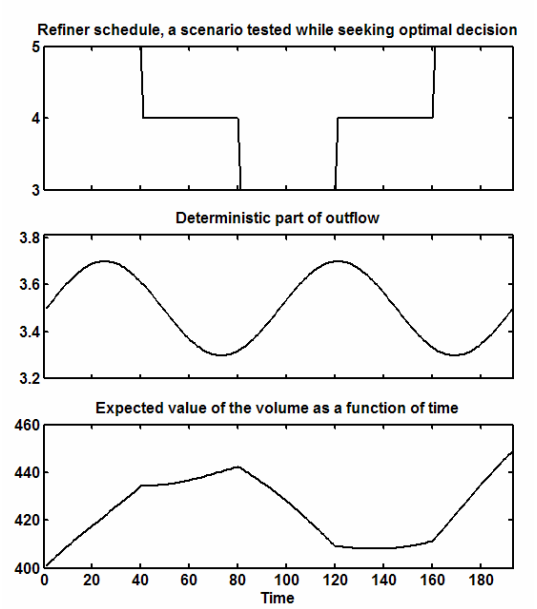


Figure 2. An example of the optimization scenario.

When stochasticity is added to the model, the value of the volume can no longer be predicted accurately. Figure 3 reveals the dramatic change in the shape of the probability distribution towards the end of the time horizon.

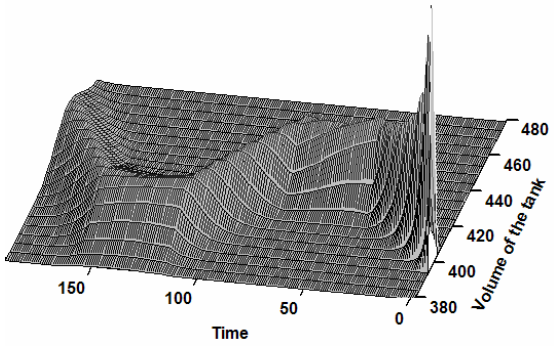


Figure 3. Probability distribution of the volume.

Adding stochasticity to the model increases the amount of information but also raises new questions: "How to deal with it?"

Random breaks on the paper machine

There are several reasons for the breaks on the paper machine and numerous ways to predict them. Yet, none of the predictions is above the others. In this paper we concentrate on the process where the TMP plant feeds one single grade paper machine. As the break occurs, there is no demand for the TMP. This kind of ongoing on / off situation can be defined with the Markov chain. Mathematically the definition goes as follows.

Suppose that $M = \{X_n\}_{n=0}^{\infty}$ is a sequence of correlated random variables, where each X_n comes from some set Ω, called the state space. We assume that states Ω in can be labeled by the integers, i.e., Ω is discrete. The process is a Markov chain if it satisfies the Markov condition

$$\mathrm{P}\left(X_{n+1} = j \mid X_n = i, X_{n-1} = x_{n-1}, \ldots, X_0 = x_0\right) = \mathrm{P}\left(X_{n+1} = j \mid X_n = i\right). \tag{5}$$

Fixing an initial distribution $\mathrm{P}(X_0 = i)$ for X_0 and the transition probability for X_{n+1} given X_n, $\mathrm{P}(X_{n+1} = \mathrm{j} | X_n = i)$ determines a Markov chain.

In this case a simple two-state Markov process is sufficient. Figure 4 illustrates the process. The process has to states, on (1) and off (0). The probability for break during next interval, if the machine is currently running is P_1, and the probability for recovering from break, if the web is currently broken is P_2.

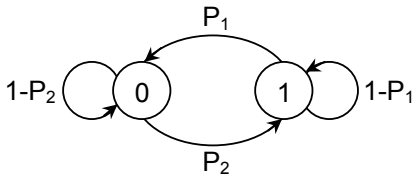


Figure 4. A two-state Markov chain.

Figures 5 and 6 show how the uncertainty affects the volume flown out. P_1 is assumed to be 0.05 and P_2 0.5. The initial state is on, the paper machine is running.

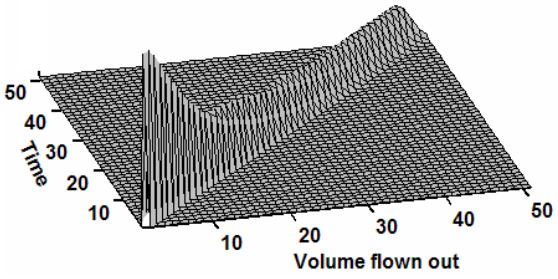


Figure 5. The evolvement of the probability distribution

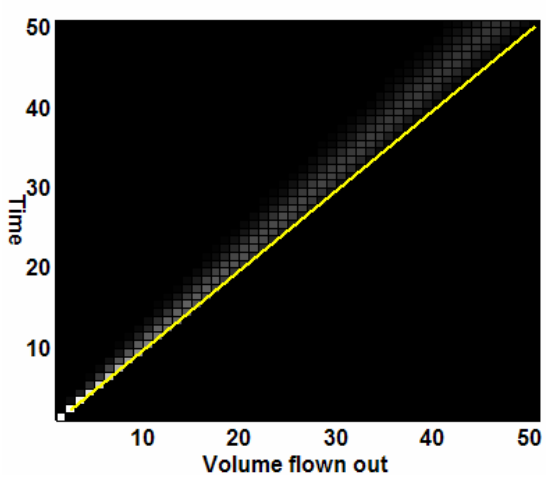


Figure 6. The average drifts away from the predicted

Stochastic objective functions and constraints

All production systems have an element of unpredictability, or stochasticity. Therefore, we cannot predict the value of our objective as a result of a set of actions precisely, but it will have a probability distribution. As far as the variance of the distributions resulting from two set of actions is roughly the same, it is sufficient to compare the mean values of objective. However, rather often the variances will be different. This is illustrated in Figure 7 where the probability distributions of grade change time (objective to be minimized) are shown for two operational practices. One distribution has a lower mean (16 minutes) but nonvanishing probability for over 20 minutes grade change time and the other has higher mean (18 minutes) but is quite predictable below 20 minutes.

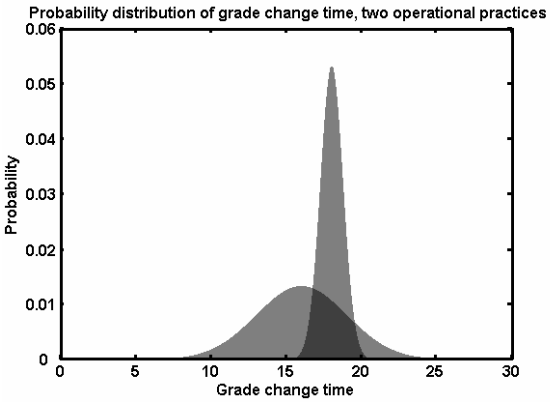


Figure 7. Two grade change options with uncertainty in the outcome.

Should the operator / engineer choose option 1 with lower average grade change time but higher risk for high grade change time, or the one with higher average, but quite predictable grade change time?

This decision is entirely subjective and expresses the decision maker's (or organization's) attitude towards risk. Quite often a systematic attitude towards risk can be expressed as a "cost of risk" or as a risk premium added to the objective.

The objective of the example case is to minimize overall electricity costs while taking into account the limited tank volumes between TMP plants and paper machines. Stochasticity can be present in two ways, as presented in the previous chapter. In the both cases the stochasticity is in the constraints.

In the deterministic case, the feasibility of a solution can easily be determined. In the TMP case the volume of the intermediate tank has upper and lower limits. If these constraints are violated, the solution is infeasible. However, this approach is not applicable, if the constraints are stochastic. In the stochastic case the solution has several different constraint values varying around some mean value. If the mean constraint value is close to the constraint boundary, the constraint values will be both bigger and smaller than the constraint boundary. Therefore, the same solution can be considered infeasible or feasible.

For stochastic systems, the end user must decide whether the constraints are absolute (the probability of violation a constraint is zero) or soft (the probability of violation is less than user specified probability $p > 0$). Equipment and safety constraints in the early part of decision horizon must be absolute, but at a later part soft: this will leave time to react if the evolution of the system is undesirable. Constraints based on operational policies are soft for stochastic systems.

Conclusions

Essentially all production decision support systems neglect the stochasticity. This is the most limiting factor of their applicability. We believe that decision support for stochastic systems with applications to production management is an interesting and highly relevant field, but yet at a research phase. One reason for practical implementations not having attacked large problems is the rising of computational costs as the number of decision variables increases.

In this paper we have taken steps to include stochastic elements to the simulation and optimization. The presented stochastic elements and the ways to deal with them are ingredients of methodology needed in designing practical implementations. Future work includes further development of methods and more complex simulation studies.

Acknowledgements

This work has been carried out as part of the DOTS (G1RD – CT – 2002 – 00755) project. We would like to express our gratitude towards European Commission and DOTS Consortium. The other members of consortium are: Papiertechnische Stiftung (PTS), Mälardalen University, Department of Public Technology, Centre Technique de l'Industrie des Papiers (CTP), Papeterie de Lancey, Papeterie de Condat, Drewsen Spezialenpapiere, Swedish Institute of Computer Science (SICS), and UPM-Kymmene.

References

[1] Pulkkinen P., Ritala R., Tienari M., Mosher A., "Methodology for dynamic optimisation based on simulation", Simulation and Process Control for the Paper Industry, Munich 2004, PTS Manuscript PTS-MS 441.

[2] Ritala R., Belle J., Holmström K., Ihalainen H., Ruiz J., Suojärvi M., Tienari M., "Operations Decision Support based on Dynamic Simulation and Optimization", Pulp and Paper Conference, Helsinki 2004.

[3] Abspoel S., Etman L., Vervoort J., Rooda J., "Simulation optimization of stochastic systems with integer variables by sequential linearization", Proceedings of the 2000 Winter Simulation Conference

[4] Dinwoodie I. H., "Stochastic simulation on integer constraint sets", SIAM J. Optim. Vol. 9, No. 1, pp. 53-61

Energy Systems

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

CONTROL VOLUME BASED MODELLING OF COMPRESSIBLE FLOW IN RECIPROCATING MACHINES

Stig Kildegård Andersen* **and Henrik Carlsen**
Department of Mechanical Engineering
Per Grove Thomsen
Informatics and Mathematical Modelling
Technical University of Denmark
DK-2800 Lyngby, Denmark

Abstract

An approach to modelling unsteady compressible flow that is primarily one dimensional is presented. The approach was developed for creating distributed models of machines with reciprocating pistons but it is not limited to this application. The approach is based on the integral form of the unsteady conservation laws for mass, energy, and momentum applied to a staggered mesh consisting of two overlapping strings of control volumes. Loss mechanisms can be included directly in the governing equations of models by including them as terms in the conservation laws. Heat transfer, flow friction, and multidimensional effects must be calculated using empirical correlations; correlations for steady state flow can be used as an approximation. A transformation that assumes ideal gas is presented for transforming equations for masses and energies in control volumes into the corresponding pressures and temperatures to improve the scaling of variables and to simplify initial conditions. Artificial dissipation for dissipating unwanted acoustic phenomena is discussed, and an asymmetric interpolation method with filtering properties for minimising numerical diffusion without introducing non physical oscillations is presented. The capabilities of the modelling approach are illustrated with a solution to a Stirling engine model that provides results in good agreement with experimental data.
Keywords: Simulation, compressible flow, one dimensional, control volume.

Nomenclature

$A_{c,i}$	Cross section at centre of vol. i [m^2]
$A_{c,ref,i}$	Reference cross section in vol. i [m^2]
$A_{ht,i,k}$	Heat transfer area of surface segment k in vol. i [m^2]
c_v	Spec. heat at const. vol. [$J/(mol \cdot K)$]
$\Delta p_{f,i}$	Frictional pressure loss in vol. i [Pa]
$h_{conv,i,k}$	Average convective heat transfer coefficient between gas and surface segment k in vol. i [$W/(m^2 \cdot K)$]
$h_{gas,i}$	Enthalpy at centre of vol. i [J/mol]
E_i	Total energy in control vol. i [J]
$F_{AD,i}$	Artificial dissipation force in vol. i [N]
$F_{wall,i}$	Wall friction force in vol. i [N]
l_i	Length of vol. i [m]
M	Molar mass of gas [kg/mol]
$\tilde{\mu}_1$	Artificial dissipation coefficient [s/m]
$\tilde{\mu}_2$	Artificial dissipation coefficient [s/m]
n_i	Number of moles in vol. i [mol]
$\dot{n}_i$	Molar rate of flow at centre of vol. i relative to volume i [mol/s]
p_i	Pressure at centre of vol. i [Pa]
p_{mean}	Mean pressure in computational domain [Pa]
$\tilde{p}_i$	Interpolated pressure at area change in vol. i [Pa]
R	Universal gas constant [$J/(mol \cdot K)$]
ρ_i	Gas density at centre of vol. i [kg/m^3]
U_i	Internal energy in vol. i [J]

* Corresponding author: Phone: +45 4525 4163, E-mail: ska@mek.dtu.dk

$v_{m,i}$	Molar spec. vol. at centre of vol. i $[m^3/mol]$
V_i	Size of vol. i $[m^3]$
$\bar{V}_i$	Velocity at centre of vol. i relative to x-coordinate system [m/s]
$\bar{V}_{CS}$	Velocity of x-coordinate system relative to inertial coordinate system $[m/s]$
T_i	Temperature at centre of volume i $[K]$
$T_{w,i,k}$	Wall temperature of surface segment k in vol. i $[K]$
x	Axial space coordinate $[m]$

Introduction

Simulation and optimisation of many machines with reciprocating pistons, such as compressors, engines and heat pumps, are at a threshold. Many machines in use today have already been optimised using analytical models or classical numerical models. In these models significant simplifying assumptions are made about the fluid dynamics and thermodynamics of the working fluid in the machines in order to reduce the complexities of the models and the efforts required to obtain solutions. But the simplifying assumptions often limit the ability of the models to correctly predict machine performance and hence reduce the likelihood that the optimum design parameters for the machines can be found using the models. New modelling approaches with a more accurate prediction of machine performance are thus needed to further optimise the machines. The new modelling approaches have to remain practical with respect to the computer power required to perform design optimisation. In this paper we present a control volume based formulation suitable for making distributed models of unsteady compressible flow that is primarily one dimensional. The method was developed for use in Stirling engine models where it has been successfully applied [1].

We work with Stirling engines, and hence we present the modelling approach with Stirling engine simulation as background. However, the approach is not Stirling specific, and we believe it is applicable in other fields of engineering.

Related methods

The choice of modelling approach for compressible flow in a reciprocating machine depends on the type and geometry of the machine to be modelled and on the phenomena that need to be resolved. It is also important to remember that steady state solutions, that are periodic in nature due to the reciprocating piston movements in the machines, are usually needed to optimise the steady state performance of the machines. Steady state periodic solutions can be found either by successively simulating a sufficient number of revolutions or by using specialised numerical methods, such as shooting or collocation, for solving boundary value problems. The number of revolutions that must be simulated to find a periodic steady state solution depends on the model, numerical method, and initial guess; including the heat capacity of the structural material of the machines in models is likely to cause slow convergence towards periodic steady state.

If multidimensional phenomena must be resolved then two- or three dimensional models are needed. Significant efforts have been put into applying software developed for computational fluid dynamics (CFD) to model reciprocating machines. CFD can be a powerful tool but, as explained in the *Best Practice Guidelines* of ERCOFTAC [2], it is very computationally intensive if accurate prediction of heat transfer and friction is needed and if the accuracy of solutions must be verified. In a recent study by Mahkamov & Djumanov [3] the commercial CFD code *Fluent* was applied to model the internal gas circuit of a Stirling engine. The gas in the engine was modelled using a relatively coarse mesh. Constant surface temperatures were prescribed in tubular- and porous matrix heat exchangers where empirical correlations were used to model heat transfer. The remaining internal surfaces of the engine were assumed adiabatic. Computations were performed on a cluster of three 2.8 GHz Dual CPU PCs and the simulations lasted approximately a day pr. revolution. The thermal inertia of the steel in the engine was not included and only a handful of revolutions were needed to approach periodic steady state conditions to the accuracy desired in the study.

Before multidimensional CFD is chosen for design optimisation it should be carefully considered if the time required for such computations is acceptable and if CFD will actually deliver any needed information that cannot be obtained with faster methods. When it is possible to use one

dimensional representations of the working volumes of, for example, reciprocating machines then models can be formulated that can be solved for steady state solutions in minutes or even seconds.

One dimensional models provide little information about transverse gradients and hence they must rely on empirical correlations and/or approximating temperature- and velocity profiles for calculating heat transfer and flow friction. Many Stirling machines can be modelled using one dimensional models because the governing phenomena in such machines, i.e. heat transfer and pressure losses, occur in flow channels that are long compared to their diameters. A traditional approach, described by Uriele & Berchowitz [4], where the effects of different loss mechanisms are added as correction terms to the calculated performance of idealised machines, is still in use in many Stirling engine models. This approach requires that the losses are decoupled from each other and they do not have significant impact on the thermodynamic cycle of the machines; these assumptions can be hard to justify and hence it is desirable to use model formulations where loss mechanisms are included directly in the governing equations.

There appears to be two main approaches for formulating one dimensional models: Either to discretise the differential forms of the one dimensional governing equations for fluid flow or to apply the integral form of the equations to a string of control volumes. The first approach has been employed, for instance, in the work of Gedeon [5][6] where the discretisation in space is combined with a temporal discretisation so that the resulting finite difference equations can be solved directly for a periodic steady state solution. Applying the integral form of the equations, i.e. applying the balance equations for mass, energy, and momentum, has the advantage that the resulting equations are meaningful regardless of the shapes of the control volumes and how the volumes are connected; when discretisation of the differential forms of the equations is performed mathematical attention must be paid to flow area discontinuities in order to maintain a meaningful formulation, as discussed by Bauwens [7].

The method presented here

The modelling approach presented here is based on the integral form of the conservation laws for mass, energy, and momentum applied to a staggered mesh consisting of two overlapping strings of control volumes. Conservation of mass and energy is applied to one string and conservation of momentum is applied to the second string. The control volumes to which conservation of momentum is applied are located so that their centres coincide with the boundaries of the control volumes where conservation of mass and energy is applied. This is the common approach of the finite volume method of CFD, and it assures direct coupling between pressure and velocity and hence yields stable and bounded solutions. The resulting formulation is conservative with respect to mass, energy and momentum. Loss mechanisms are included directly in the governing equations by including them in the balance equations. A transformation is presented that uses the ideal gas equation of state to obtain a well scaled system of first order ordinary differential equations (ODEs) to which realistic initial values can easily be provided. Empirical correlations and correction factors for steady state flow, that are readily available in the literature, can be used for estimation of heat transfer, flow friction and multi dimensional effects. Asymmetric interpolation methods with filtering properties are used for interpolating temperatures at control volume boundaries in order to minimise the numerical diffusion, that can accompany advective energy transport, without introducing non-physical oscillations in solutions. Because the approach results in models that describe fully compressible flow, solutions to the models may include acoustic phenomena. If acoustic phenomena are of little interest they may represent a large and unwanted computational overhead if they oscillate at significantly higher frequencies than the rotational frequencies of the reciprocating machine. For this reason artificial dissipation can be included in the momentum balance to quickly dissipate such acoustic waves.

This paper contains an overview of the modelling approach and shows an example solution to a Stirling engine model created using the formulation as illustration.

Method

Discretisation

The gas filled domain is discretised into a one dimensional staggered mesh of control volumes as shown in Figure 1. In the following ODEs for the pressure p_i and temperature T_i at the centre of a solid line control volume and an ODE for the velocity $\bar{V}_j$ at the centre of a dashed line control volume are derived.

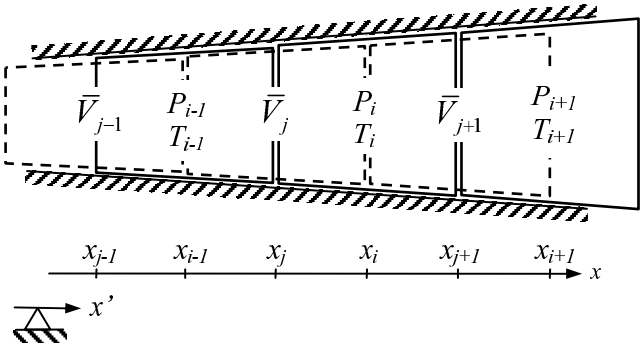


Figure 1: A staggered mesh of control volumes

The mass and energy balances

The mass balance, on molar basis, for volume *i* is written as follows:

$$\frac{dn_i}{dt} = \dot{n}_j - \dot{n}_{j+1} = \underbrace{\frac{\bar{V}_j \cdot A_{c,j}}{v_{m,j}} - \frac{\bar{V}_{j+1} \cdot A_{c,j+1}}{v_{m,j+1}}}_{Advection} \quad (1)$$

In writing the energy balance for control volume *i* it is assumed that the kinetic and gravitational potential energies in the control volume are of negligible magnitudes so that the total energy E_i in control volume *i* equals the internal energy U_i.

$$\frac{dE_i}{dt} = \frac{dU_i}{dt} = \underbrace{\dot{n}_j \cdot \left(h_{gas,j} + \frac{M}{2} \cdot \bar{V}_j^2 \right)}_{Advection} - \underbrace{\dot{n}_{j+1} \cdot \left(h_{gas,j+1} + \frac{M}{2} \cdot \bar{V}_{j+1}^2 \right)}_{Advection} + \underbrace{\sum_k h_{conv,i,k} \cdot A_{ht,i,k} \cdot \left(T_{w,i,k} - T_i \right)}_{Convective\ heat\ transfer\ from\ walls} - \underbrace{p_i \cdot \frac{dV_i}{dt}}_{Volume\ change} \quad (2)$$

Here only changes in U_i due to advection, convective heat transfer, and volume change are included; Additional mechanisms, such as axial heat conduction or internal heat production, that affect the energies in the control volumes may be included as additional terms if needed. Note that the gas in volume *i* may exchange heat with more than one surface.

Transformation of mass and energy balances

Models can be formulated directly using the mass and energy balances in (1) and (2) but doing so may not be optimal. If the computational domain contains control volumes of different sizes then the masses and energies in the volumes will also differ even if the pressure and temperature are constant throughout the domain. If the sizes of the control volumes differ by orders of magnitudes then the corresponding differences in the sizes of the masses and energies can make it more difficult to accurately solve the ODEs for the masses and energies due to bad scaling of the variables. A transformation of the ODEs for masses and energies into ODEs for pressures and temperatures is desirable because it can improve the scaling of the variables and at the same time make it easier for an engineer to provide realistic initial values to a model. For an ideal gas, using $p \cdot V = n \cdot R \cdot T$, $U = n \cdot u$, and $du = c_v(T) \cdot dT$, gives:

$$\frac{dT_i}{dt} = \frac{1}{n_i \cdot c_v(T_i)} \cdot \left(\frac{dU_i}{dt} - \frac{U_i}{n_i} \cdot \frac{dn_i}{dt} \right) \quad (3)$$

$$\frac{dp_i}{dt} = p_i \cdot \left(\frac{1}{n_i} \cdot \frac{dn_i}{dt} + \frac{1}{T_i} \cdot \frac{dT_i}{dt} - \frac{1}{V_i} \cdot \frac{dV_i}{dt} \right) \quad (4)$$

The transformation defined by (3) and (4) can be used with the derivatives calculated in (1) and (2) as input. The transformation can be enhanced by scaling the units of the pressures and temperatures between the model and the numerical method. (3) and (4) require unscaled units but the numerical method may integrate the pressures and temperatures in units of, say, MPa and hK (hekto-Kelvin) if this helps to achieve better scaling of the variables.

The momentum balance

The momentum balance for the dashed line control volume *j* is written as (5).

$$\frac{d\left(\mathrm{M}\cdot n_j\cdot\bar{V}_j\right)}{dt}=$$
$$\underbrace{p_{i-1}\cdot A_{c,i-1}-p_i\cdot A_{c,i}+\tilde{p}_j\cdot\left(A_{c,i}-A_{c,i-1}\right)}_{\text{Pressure forces}}$$
$$+\underbrace{\mathrm{M}\cdot\left(\frac{\bar{V}_{i-1}{}^2\cdot A_{c,i-1}}{v_{m,i-1}}-\frac{\bar{V}_i^2\cdot A_{c,i}}{v_{m,i}}\right)}_{\text{Advection}}$$
$$-\underbrace{\frac{\bar{V}_j}{\left|\bar{V}_j\right|}\cdot\left|F_{wall,j}\right|}_{\text{Wall friction}}+\underbrace{F_{AD,j}}_{\text{Artificial dissipation}}$$
$$-\underbrace{\mathrm{M}\cdot n_j\cdot\frac{d\bar{V}_{CS}}{dt}}_{\text{Acceleration of coordinate system}} \qquad (5)$$

By expanding the accumulation term on the left hand side in (5) the time derivative of the velocity in volume *j* becomes:

$$\frac{d\bar{V}_j}{dt}=\frac{1}{\mathrm{M}\cdot n_j}\cdot\left\{\right.$$
$$\underbrace{p_{i-1}\cdot A_{c,i-1}-p_i\cdot A_{c,i}+\tilde{p}_j\cdot\left(A_{c,i}-A_{c,i-1}\right)}_{\text{Pressure forces}}$$
$$+M\cdot\left(\frac{\bar{V}_{i-1}\cdot A_{c,i-1}}{v_{m,i-1}}\cdot\left(\bar{V}_{i-1}-\bar{V}_j\right)\right.$$
$$\underbrace{\left.-\frac{\bar{V}_i\cdot A_{c,i}}{v_{m,i}}\cdot\left(\bar{V}_i-\bar{V}_j\right)\right)}_{\text{Advection and accumulation}}$$
$$\left.-\underbrace{\frac{\bar{V}_j}{\left|\bar{V}_j\right|}\cdot\left|F_{wall,j}\right|}_{\text{Wall friction}}+\underbrace{F_{AD,j}}_{\text{Artificial dissipation}}\right\}$$
$$-\underbrace{\frac{d\bar{V}_{CS}}{dt}}_{\text{Acceleration of coordinate system}} \qquad (6)$$

In the following the term $\Delta p_{f,j}$ is used to represent the absolute value of the pressure loss caused by flow friction between the inlet and outlet of volume *j*. $\Delta p_{f,j}$ can be approximated by using empirical correlations for the friction factor and loss coefficients developed for steady state flow. The term $F_{wall,j}$ in (6) can then be approximated by multiplying the pressure loss with a cross sectional reference area $A_{c,ref,j}$:

$$F_{wall,j}=\Delta p_{f,j}\cdot A_{c,ref,j} \qquad (7)$$

It appears appropriate to choose $A_{c,ref,j}$ to be equal to either $A_{c,i-1}$, $A_{c,i}$, or some mean value in between these two cross sectional areas.

To determine the pressure $\tilde{p}_j$ in (6) acting on the area difference between the inlet and outlet of volume *j*, we impose the condition that (6) must match the energy equation for steady state incompressible flow in a stationary tube, i.e. the extended Bernoulli equation, between x_{i-1} and x_i when gravitational potential energies are neglected:

$$\left(p_{i-1}+\rho_j\cdot\frac{\bar{V}_{i-1}{}^2}{2}\right)-\left(p_i+\rho_j\cdot\frac{\bar{V}_i^2}{2}\right)=\frac{\bar{V}_j}{\left|\bar{V}_j\right|}\cdot\Delta p_{f,j} \qquad (8)$$

It can be verified by insertion that this is achieved if the pressure $\tilde{p}_j$ is calculated using either (9) or (10) and the artificial dissipation force is zero. Note that the last terms in (9) and (10) simplify greatly if $A_{c,ref,j}$ is chosen equal to either $A_{c,i-1}$ or $A_{c,i}$.

$$\tilde{p}_j=p_i+\frac{1}{2}\cdot\rho_j\cdot V_i^2\cdot\left(1-\frac{A_{c,i}}{A_{c,i-1}}\right)$$
$$+\frac{\frac{A_{c,ref}}{A_{c,i-1}}-1}{\frac{A_{c,i}}{A_{c,i-1}}-1}\cdot\frac{\bar{V}_j}{\left|\bar{V}_j\right|}\cdot\Delta p_{f,j} \qquad (9)$$

$$\tilde{p}_j = p_{i-1} + \frac{1}{2}\cdot\rho_j\cdot\bar{V}_{i-1}{}^2\cdot\left(1-\frac{A_{c,i-1}}{A_{c,i}}\right) - \frac{1-\frac{A_{c,ref,j}}{A_{c,i}}}{1-\frac{A_{c,i-1}}{A_{c,i}}}\cdot\frac{\bar{V}_j}{\left|\bar{V}_j\right|}\cdot\Delta p_{f,j} \qquad (10)$$

(9) and (10) give $\tilde{p}_j$ as the pressure in a solid line control volume plus two correction terms. At flow area discontinuities, such as the location where a tube connects to a large chamber, the correction terms containing the square of the velocity can cause instability. To avoid instability the choice between (9) and (10) should be made so that the squared velocity is as small as possible, i.e. so that the extrapolation from the pressure in the solid line control volume is as small as possible. In practice this can be done by using (9) where $A_{c,i} \geq A_{c,i-1}$ and using (10) elsewhere.

Artificial dissipation

The artificial dissipation force $F_{AD,j}$ is included to quickly dissipate acoustic waves. This has been useful when simulating Stirling engines where there are no valve movements or internal explosions that generate acoustic waves that are important to engine performance. The acoustic waves generated by, for instance, poor initial values represent an unwanted computational overhead. We use (11) for calculating the artificial dissipation force.

$$F_{AD,j} = \left\{-\tilde{\mu}_1\cdot l_j\cdot\left(A_{c,i-1}\cdot\left.\frac{\partial\bar{V}}{\partial x}\right|_{i-1} - A_{c,i}\cdot\left.\frac{\partial\bar{V}}{\partial x}\right|_{i}\right) + \tilde{\mu}_2\cdot l_j{}^3\cdot\left(A_{c,i-1}\cdot\left.\frac{\partial^3\bar{V}}{\partial x^3}\right|_{i-1} - A_{c,i}\cdot\left.\frac{\partial^3\bar{V}}{\partial x^3}\right|_{i}\right)\right\}\cdot p_{mean} \qquad (11)$$

The terms in (11) proportional to the first derivative of the velocity can be considered similar to a viscous normal stress [8]. The terms proportional to the third derivatives of the velocity cannot easily be linked to physical effects. The terms proportional to the third spatial derivatives of the velocity inhibit oscillations between neighbouring pairs of control volumes, and hence dampen oscillations with a short wavelength compared to the length of the computational domain. The terms proportional to the first spatial derivatives of the velocity penalize curved velocity profiles and hence dampen oscillations with longer wavelengths. The artificial dissipation force is scaled by a mean pressure that should be representative for the region where the artificial dissipation is applied. In our simulations of Stirling engines we have used the time varying average pressure in the engine for the scaling.

Care must be taken not to apply the artificial dissipation forces so that they interfere significantly with the mean solution. Therefore the artificial dissipation coefficients should not be too large and (11) should not be used where flow area discontinuities, localised pressure losses, or similar affect the velocity derivatives in (11). We have not observed problems from simply switching the artificial dissipation terms off, when any of the dashed line control volumes containing the velocities used for calculating the derivatives in (11) contain flow area changes or are prescribed localised pressure losses to model inlets, tube bends, or similar.

The magnitudes of the artificial dissipation coefficients $\tilde{\mu}_1$ and $\tilde{\mu}_2$ must be large enough to yield the desired smoothing effect and yet small enough to not significantly affect the mean solution. We test for the first by visual inspection of solutions and for the latter by inspecting the ratio $\left|F_{AD,j}/F_{wall,j}\right|$. In our simulations of Stirling machines with Helium, Nitrogen, or air at pressures in the range of 1-10 MPa we use the values $\tilde{\mu}_1 = 1.0\cdot10^{-4}\,s/m$ and $\tilde{\mu}_2 = 2.0\cdot10^{-6}\,s/m$.

Interpolation of state variables and velocities

The pressures p_j and temperatures T_j at the centres of the dashed line control volumes and the velocities $\bar{V}_i$ at the centres of the solid line control volumes are used both explicitly and implicitly, through gas property calculations, in equations (1) through (10). Since the formulation does not yield these values directly they must be approximated from known values. We use interpolation for the

approximations to increase the accuracy, i.e. the order, of the spatial discretisation, to reduce the number of control volumes needed to obtain accurate solutions.

p_j and $\bar{V}_i$ are interpolated using interpolating polynomials through known values distributed symmetrically around the point of interpolation; we use linear interpolation next to the boundaries of the computational domain and next to flow area discontinuities and cubic interpolation elsewhere. T_j, however, is interpolated using asymmetric interpolation methods with filtering properties in order to avoid the well known problem of non-physical oscillations in solutions due to the discretisation. Kühl and Schultz [9] presented an interpolation method based on a blend between symmetric linear interpolation and linear extrapolation through two upstream points. In our experience the oscillations are better reduced by an interpolation method based on a cubic polynomial that goes through one upstream point and two downstream points and whose first derivative at the upstream point equals the slope of the tangent through two upstream points, as illustrated in Figure 2. If the four solid line control volumes closest to x_j are of equal length then T_j can be interpolated in this way with the stencil (12). We do not use this interpolation method across flow area discontinuities or at the ends of the computational domain. Here we resort to either the interpolation method by Kühl and Schultz or to symmetric linear interpolation depending on how many neighbouring values are available for the interpolations.

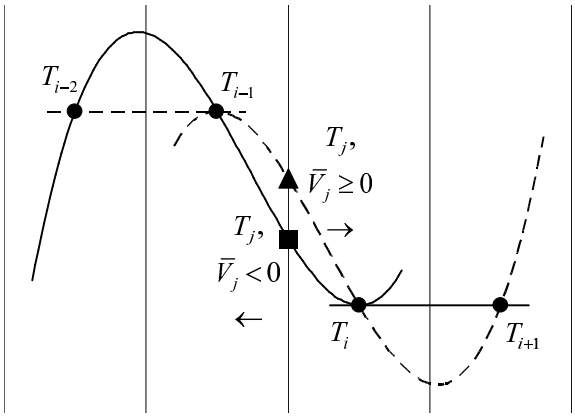


Figure 2: Asymmetric cubic interpolation

$$T_j = \begin{cases} \frac{1}{32}\cdot\left(-6\cdot T_{i-2} + 27\cdot T_{i-1} + 12\cdot T_i - T_{i+1}\right), \bar{V}_j \geq 0 \\ \frac{1}{32}\cdot\left(-T_{i-2} + 12\cdot T_{i-1} + 27\cdot T_i - 6\cdot T_{i+1}\right), \bar{V}_j < 0 \end{cases} \quad (12)$$

Example: A Stirling engine model

To illustrate the capabilities of the modelling approach we present a solution to a one dimensional Stirling engine model created using the formulation. For illustration purposes engine specific details have little relevance as do many details specific to the model; the following introduction is thus a short one. A more detailed description of the model and engine along with an experimental validation of the model can be found in [1].

The working volume of a Stirling engine consists primarily of a cold and a hot cylinder volume and a serial connection of heat exchangers, viz. a cooler, a regenerator, and a heater, that connect the cylinder volumes. The cooler and heater are tubular heat exchangers and the regenerator is a void filled with a porous matrix with a large heat transfer area. During operation the sizes of the cylinder volumes vary in a periodic fashion so that that the gas is alternately compressed, pushed towards the hot cylinder volume so that the gas is heated, expanded, and pushed towards the cold cylinder volume so that the gas is cooled. The heating and cooling of the gas influences the pressure in the engine and when the expansion occurs at a higher average pressure than the compression then work can be extracted from the engine cycle. During operation a large temperature difference builds up across the regenerator and the regenerator matrix then acts as a thermal heat storage. When gas is flowing towards the cooler the matrix absorbs energy from the gas and when the gas flow is reversed the gas absorbs the deposited energy from the matrix. In this way the regenerator minimises the amount of energy that is carried from the heater to the cooler by the working gas.

A distributed model of a Stirling engine is desirable because of the large spatial and temporal variations in the properties and velocities of the working gas internally in the components of the engine. The discretisation of the working volume

used in the Stirling engine model, i.e. the computational domain, is illustrated in Figure 3.

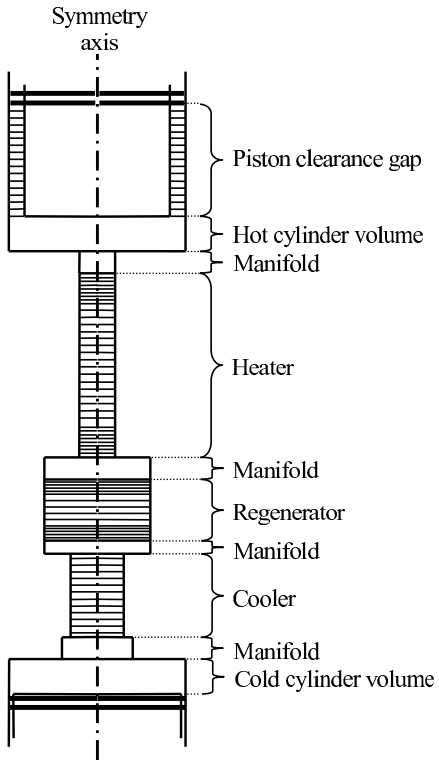


Figure 3: Discretisation in Stirling engine model

The domain contains the major components of a Stirling engine along with some manifold volumes and a long thin gap that exits in the clearing between the piston and cylinder wall in the hot cylinder volume. The model is made so that the cylinder volumes and manifold volumes are lumped into single volumes whereas the cooler, regenerator, heater, and piston clearance gap are represented by strings of control volumes so that the effects of axial variations in gas properties and velocities in these components are included in solutions to the model. The discretisation can be locally refined where large gradients exist in the solutions. Where parallel flow paths exist, as in the tubular heat exchangers, they are lumped together into single flow paths in the discretisation; the exact geometry is only taken into account when calculating heat transfer and flow friction. The control volumes in the piston clearance gap follow the motion of the piston and all other control volumes are stationary in space. The steel in the engine is also discretised and modelled so that wall temperatures and heat conduction in the walls can be determined. The model thus deals with reversing flow and large pressure and temperature oscillations in a computational domain with both stationary and moving control volumes, and with the coupled thermodynamics of the gas and the steel in the engine. The pressure in the engine goes as high as 10 MPa, but the temperatures are such that the inaccuracy introduced by the ideal gas assumption in (3) and (4) is just acceptable when compared to other inaccuracies in the model. The example solution presented below was computed using the in house software *MusSim* (*Multi Purpose Software for Simulation*).

Results and discussion

To illustrate the capabilities of the modelling approach presented above Figure 4 shows how the pressures, pressure losses, temperatures, and velocities vary in time in a solution to a Stirling engine model created using the modelling approach.

In Figure 4 the topmost plots shows the variations in time of the pressure and the shape of the computational domain, but it is difficult to see any spatial pressure variations. The second plot shows the difference between the pressure in different areas of the domain and the pressure in the cold cylinder volume. The pressure difference changed sign twice in the cycle because the flow direction of the gas in the serial connection of heat exchangers changed direction twice. It can be seen that the pressure gradients became larger in the porous matrix of the regenerator than anywhere else in the engine.

The effects of inlet pressure losses are also visible in the plot of the pressure differences in Figure 4. The clearest example is at the inlet from the hot cylinder volume to the heater at approximately 90 degrees crank angle; here a pressure drop was caused by the acceleration of the gas from almost stationary in the hot cylinder volume to the flow speed in the heater and by an inlet pressure loss coefficient that was applied at the inlet. Wave phenomena are most visible in the plot near the hot cylinder volume in the last quarter of the revolution. They were induced when flow reversal caused a momentary dip in the powerful heat exchange predicted by the empirical correlations used in the model. Note that the visible waves are in the time wise direction; no waves can be seen travelling along the domain.

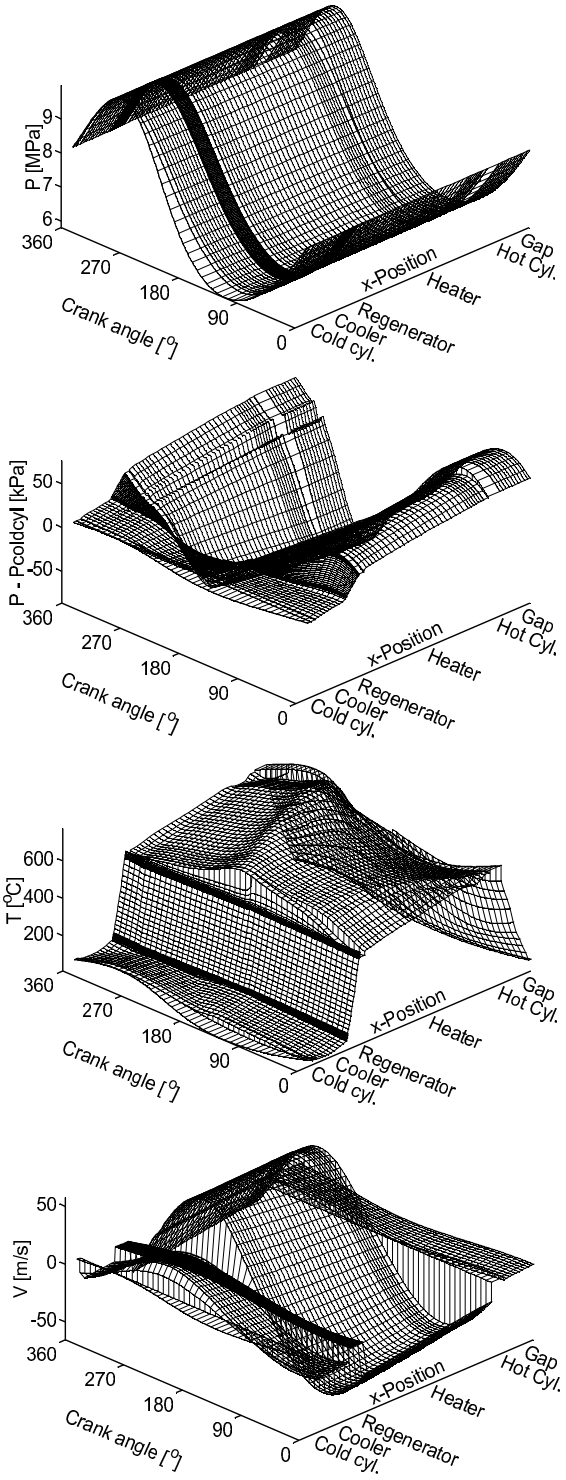


Figure 4: Variations in space and time (plotted as crank angle) of pressures, pressure losses, temperatures, and velocities in a solution to a Stirling engine model

The third plot from the top in Figure 4 shows the large variations in temperature in the domain caused by the volume changes of the domain and by heat exchange. Steep temperature gradients can be seen in the regenerator and in the hot piston gap where numerical diffusion was minimised by the interpolation methods used in the model. The temperature variations in the regenerator are small due to the powerful heat exchange and the large heat capacity of the regenerator matrix.

The bottom plot in Figure 4 shows the flow velocities in the domain. The velocities are plotted as positive when the flow was towards the hot cylinder volume. Abrupt changes in velocity are clearly visible at the area discontinuities between the components. The largest velocities can be seen in the heater of the engine where it is just visible that velocity changes travelled along the domain; they did not occur simultaneously throughout the domain.

Conclusion

A modelling approach for making one dimensional models of compressible flows has been presented. The capabilities of the approach have been illustrated with an example solution to a distributed Stirling engine model created using the approach.

Acknowledgements

The initial development of the *MusSim* software was sponsored by the gas company DONG.

References

[1] Andersen S K, Carlsen H, Thomsen P G. S*imulation of Temperature Fluctuations in Stirling Engine Regenerator Matrices*, In: Proc. of the 11th International Stirling Engine Conference, Rome, 2003, pp. 120-127

[2] Casey M, Wintergate T (editors). *Best Practice Guidelines.* European Research Community On Flow, Turbulence And Combustion - Special Interest Group on "Quality and Trust in Industrial CFD", 2000.

[3] Mahkamov K, Djumanov D. *Three-Dimensional CFD Modelling of a Stirling Engine*. In: Proc. of the 11th International Stirling Engine Conference, Rome, 19th-21st November 2003, pp. 96-107

[4] Urieli I, Berchowitz D. *Stirling Cycle Engine Analysis*. Adam Hilger Ltd., Bristol, 1984.

[5] Gedeon D. *A Globally-Implicit Stirling Cycle Simulation*. In: Proc. of the 21th Intersociety Energy Conversion and Engineering Conference Vol. 1, American Chemical Society, San Diego, 1986, pp. 550-554

[6] Gedeon D. *Sage: Object Oriented Software for Stirling Machine Design*. In: Proc. of the 29th Intersociety Energy Conversion and Engineering Conference Vol. 4, American Institute for Aeronautics and Astronautics, Monterey CA, 1994, pp. 1902-1907

[7] Bauwens L. *Stirling Engine Modelling on Eulerian Grids*. In: Proc. of the 28th Intersociety Energy Conversion and Engineering Conference Vol. 2, American Chemical Society, Atlanta, 1993, pp. 2725-2730

[8] Bird R, Stewart W, Lightfoot E. *Transport Phenomena*. John Wiley & Sons, 1960

[9] Kühl HD, Schults, S. *Modelling of Thermal Regenerator Losses Due to Axial Flow Dispersion*. In: Proc. of the 7th International Conference on Stirling Cycle Machines, Tokyo, November 5-8, 1995

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

NODAL MODELLING OF BIOMASS THERMOCHEMICAL DECOMPOSITION

Razvan Andrei Vijeu*, Jerôme Bellettre and Mohand Tazerout
Ecole des Mines de Nantes
Département Systèmes Energétiques & Environnement
France

Abstract

The pyrolysis model presented in this paper consists in a coupling between a heat transfer model, a chemical model for the thermal decomposition of the pyrolysed material and a simplified mass transfer model for the resulting gaseous phase. In order to avoid the use of complicated partial differential equations systems for the description of the heat and mass transfer phenomenons a nodal method is employed. The chemistry being directly linked to the heat transfer the model allows very simple use of different types of chemical mechanisms suitable for various types of biomasses. A simplified approach for the mass transfer model is currently under construction thus only some preliminary results are presented.
Keywords: biomass, pyrolysis, modelling

Nomenclature

C_p	specific heat [J/kg/K]
k	reaction constants [s^{-1}]
K	permeability [m^2]
m	mass [kg]
$\dot{m}_{tr}$	mass flow [$kg/m^3/s$]
p	pressure [Pa]
S	surface [m^2]
t	time [s]
T	temperature [K]
u	velocity [m/s]
V_c	control volume [m^3]
x	distance [m]

Greek letters

ρ_s	apparent density [kg/m^3]
ε	bed void fraction
Φ	heat flux [W]
μ	fluid viscosity [kg/m/s]

Subscripts

C	char
g, G	gaz
s	solid
T	tar
W	wood

Introduction

As a result of the present environmental context more and more attention is given to the biomass conversion technologies as green energy sources. This fact accents the need for accurate models that can predict thermal decomposition behaviour and that can assist the scale up of new types of installations.

This work aims to investigate the opportunity of implementing a new type of model that can provide fast but accurate information regarding the thermal decomposition of biomass. The model combines a nodal numerical method, usually employed for heat transfer cases, with a chemical mechanism for the thermal decomposition and with a simplified mass transfer model. The result is an unsteady state model that can predict, depending on the material properties and the chosen chemical sub-model, the parameters of the pyrolysis process. The model also allows the study of the influence of

* Corresponding author. Phone +33 (0) 2 51 85 82 74
Fax: +33 (0) 2 51 85 82 99, E-mail: vijeurazvan@hotmail.com

different reactor geometries on the global process and, as a result, it offers the scale up possibility.

Reactor geometry

As stated earlier, the model should be able to study the influence of the reactor geometry on the overall pyrolysis process. However, it is generally admitted that the geometry of the reactor can only significantly influence its heat transfer capabilities, such the surface to volume ratio for example. Since the non oxidative pyrolysis is a mainly endothermic process it is highly dependent on the temperature, thus dependent of the amount of heat introduced into the reactor through the walls. From this point of view the advantage lies with the geometries that provide a bigger heat exchanging surface for a constant volume.
Regarding the numerical the symmetrical geometries are preferred since they are more representative for the actual industrial installations.

The numerical method

The nodal method is widely used for the heat transfer modelling of non homogenous media in steady or unsteady state. It allows the construction of simple heat transfer models by linking interacting elements and is more adequate for this type of problems comparing to the finite volumes or finite elements methods. The principles of this numerical method are discussed in [1], [2] and [3].
For better understanding all the concepts of the nodal method will be exemplified on the reaction volume presented in figure 1. It is filled with spherical biomass particles with the same diameter forming a porous bed. If the bed is considered homogenous and isotropic an annular discretization like the one shown in figure 1 can be applied. The result is a two dimensional nodal grid. However, if the zone is not homogenous and isotropic, a more complex discretization pattern has to be employed, by inserting several separation planes passing through the axis of the cylinder.
The nodal numerical method implies that the entire control volume is concentrate in a single point in the centre of the volume and the temperature is assumed uniform within a control volume.
For the annular shaped volumes the resulting nodal grid is presented in figure 2. It represents a half axial cross section of the cylinder as it is discretized in figure 1. The distance between the nodes usually varies from a few millimeters to a few centimeters. The filled points correspond to the reaction volume wile the unfilled points represent the boundaries. The interactions between a boundary point and a reaction point are defined separately due to the particularity of various phenomenons in their vicinity.

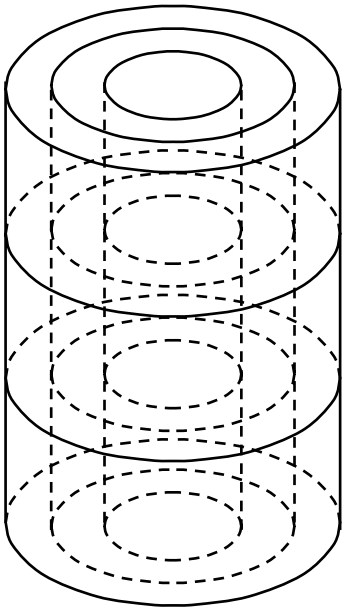


Figure 1: The reaction volume and the discretization

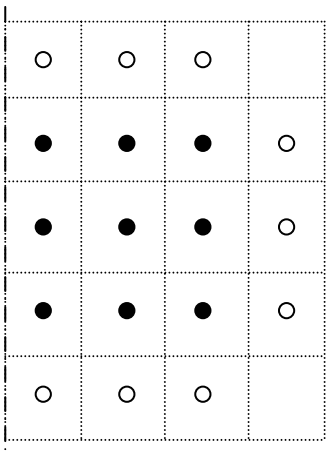


Figure 2: The nodal grid

The algorithm of the nodal method implies that the value of a certain parameter at the current time step is calculated using the available data from the previous time step. This explicit technique allows analytical solving of differential equations considering that certain parameters remain constant during the imposed time period. This

simplification eliminates the need of iterating or solving complex systems of equations and provides a fast calculation method with minimal precision loss. However, the method requires a low time step for some of the occurring phenomenons. An analysis of the time step is latter presented.

The heat transfer sub model

The heat transfer modelling of the porous bed during pyrolysis is based on the previous work presented in [4]. A number of assumptions are made in order to simplify the model. First, depending of the heating configuration and the gas flow in the reactor, a choice has to be made regarding the temperature field of the solid and fluid phases. The thermal equilibrium between the two phases is best suitable if the cylinder in figure 1 is heated only on the lateral side for example, with the gases escaping through the top side. However, if the same cylinder is also heated on the bottom side, the hot gas formed in the lower layers must pass through the colder superior layers. In this case the thermal equilibrium is no longer suitable and two temperature fields need to be employed for a better description of the heat transfer phenomenons. In this work, the latter case is retained.

The heat transfer within the fluid can also be simplified by assuming that the fluid phase exchanges heat only by convection with the solid phase, thus neglecting its own conduction and radiation. The solid phase radiation and conduction can be quantified together in a parallel mechanism, following the heat transfer mechanism proposed by the authors in [4].

The heat transfer equation applicable to the nodal model is given by [1] for a node *i* surrounded by *j* other nodes:

$$m_i C_{pi} \frac{dT_i}{dt} = \sum_j \Phi_{ij} + \sum_k \Phi_{ki} \qquad (1)$$

The term Φ_{ij} represents the flux exchange between the nodes *i* and *j* and is expressed by equation 2.

$$\Phi_{ij} = G_{ij} \cdot (T_i - T_j) \qquad (2)$$

The G_{ij} factor is the conductance between the two nodes and it is calculated in function of the material properties and in function of the reactor geometry. The Φ_{ki} flux represents the source term for the node *i* and it is a function of the heat of reaction in table 1 and the kinetic constants of the chemical process. If two temperature profiles are to be taken into account, the equation (1) can be applied for each phase and the heat flux resulted form the convection between the two phases can be included as heat source terms.

The chemical sub model

One of the advantages of this method is its versatility in the use of various chemical mechanisms. The kinetic models are best suitable for the nodal method but combinations between such models and equilibrium models are possible.
Different kinetic models are presented in [6], [7], [8] and [9]. The model presented by Mousques in [9] and shown in figure 3 is chosen to exemplify the use of kinetic mechanisms.

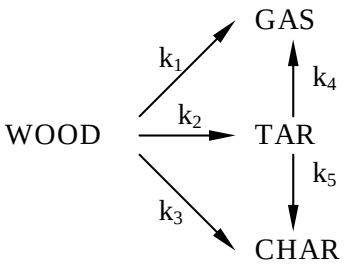


Figure 3: Chemical model

The wood is decomposed by three simultaneous primary reactions producing a solid carbonaceous residue (the char), a pseudo species of condensable gases (the tar) and a pseudo species of non condensable gases (the gas). The secondary decomposition of the tar produces more gas and char. The reaction constants obey the Arrhenius formalism and the kinetic data is presented in table 1.

Reaction	A_i (s^{-1})	E_{ai} (J/mol)	ΔH_i (J/kg)
1	$1.43\ 10^4$	$8.86\ 10^4$	$4.18\ 10^5$
2	$4.13\ 10^6$	$1.127\ 10^5$	$4.18\ 10^5$
3	$7.38\ 10^5$	$1.065\ 10^5$	$4.18\ 10^5$
4	$4.28\ 10^6$	$1.08\ 10^5$	$-4.2\ 10^4$
5	10^5	$1.08\ 10^5$	$-4.2\ 10^4$

Table 1: Kinetic data for the chemical model [8]

This chemical model supposes that all the reactions are first order decompositions. The parameters

used for the expressions of the kinetic laws are give by equations 3 and 4.

$$\rho_s = \frac{m_s}{V_c} \tag{3}$$

$$\rho_g = \frac{m_g}{\varepsilon \cdot V_c} \tag{4}$$

The two parameters represent the apparent densities of the solid or gaseous pseudo species. In the case of the solids, the density is relative to a control volume *V*,. while for the gases it is relative to the void volume available in the same control volume.
The mass balance for the four pseudo species involved in the pyrolysis process is given by equations (4) to (7).

$$\frac{d\rho_W}{dt} = -(k_1 + k_2 + k_3)\rho_W \tag{5}$$

$$\frac{d\rho_C}{dt} = k_3 \cdot \rho_W + k_5 \cdot \rho_T \tag{6}$$

$$\frac{d\rho_G}{dt} = k_1 \cdot \rho_W + k_4 \cdot \rho_T \tag{7}$$

$$\frac{d\rho_T}{dt} = k_2 \cdot \rho_W - (k_4 + k_5) \cdot \rho_T \tag{8}$$

The heat of different reactions in table 1 provide means of calculation for the Φ_{ki} terms in equation (1).
The versatility of this approach lies in the fact that for a different material, characterized by different reaction mechanism, the equations (5) to (8) are easily replaceable with those resulted from the new mechanism. However, the mass balance for the fluid phase for each control volume must include the terms describing the flow of the gasses in the reactor.

The mass transfer sub model

If the heat transfer and the chemistry are easily integrated with the nodal method concepts, the fluid flow component represents the difficulty of this method, mainly due to the transitory nature of the global process.
The first approximation to be made is to consider all the gasses as ideal. This allows the evaluation of the pressure build up in each grid node during the imposed time step. The fluid velocities can be obtained knowing the pressures in each node and using Darcy's law for porous media:

$$u = -\frac{K}{\mu}\frac{\delta p}{\delta x} \tag{9}$$

Patankar in [3] recommends defining a secondary grid. This new grid will have as nodes the interfaces between the adjacent cells and the velocities will be disposed as in figure 4. Since the pressure data is available in discrete positions, the equation (9) has to be slightly modified. Equation (10) represents a more suitable form of the Darcy law.

$$u = -\frac{K}{\mu}\frac{\Delta p}{\Delta x} \tag{10}$$

The term Δp represents the pressure difference between two neighbor nodes and Δx represents the distance between the nodes.

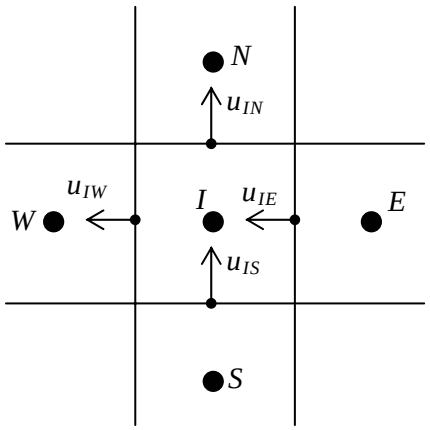


Figure 4: The velocities grid

The mass transfer between two nodes can be evaluated using the calculated velocity and the common surface of the two nodes.

$$\dot{m}_{tr} = u\frac{S}{V_c}\rho_g \tag{11}$$

Besides the convective mass transfer, a diffusional mass transfer component can also be defined using a mass transfer coefficient. In this model however, there is no need to account for this kind of mass transfer.

Coupling of the occurring phenomenons and results

As stated before, the method used to build the pyrolysis model links three sub models describing three different phenomenons. Some difficulties can arise in deciding the order in which the sub models are to be applied. The logical chain of events is:

1. the heat transfer that determines the advancement of the chemical reactions;
2. the chemical reactions caused by the transferred heat, causing a rise of the pressure in the affected nodes;
3. the flow of the gas produced by the chemical reactions, causing a redistribution of the gaseous reactants and determining a new convective heat transfer component between the gas and the solid.

The first step of the calculation chain can be one of the three, but their calculation order should be respected since it represents the causal chain of the process.

Another major difficulty of this method is the effective use of the mass transfer sub model as it is presented in the previous section. The problem consists in the fact that the gas flow phenomenon has a very small time constant compared to the heat transfer and the chemistry time constants. This problem causes unacceptable simulation delays and for this reason the mass transfer mechanism should be implemented by separating the mass flow phenomenon of the rest of the processes.

The following results are preliminary and they are calculated neglecting the occurrence of the gas flow in the reactor. While this situation is indeed far from the reality the presented results are only shown in order to illustrate the capabilities of the nodal method.

Figure 5 presents a type of result obtained for the temperature field within a reactor similar to the one in figure 1, heated on the side surface with 800K and on the bottom surface with 1000K. The main grid has 400 vertical nodes and 200 horizontal nodes, for a length of the cylinder of 40cm and a radius of 20cm. The simulation time is 1000 seconds with a time step of 0.5 seconds. Figure 6 presents the biomass apparent density field ρ_W (from equation 5) in the same reactor and in the same conditions.

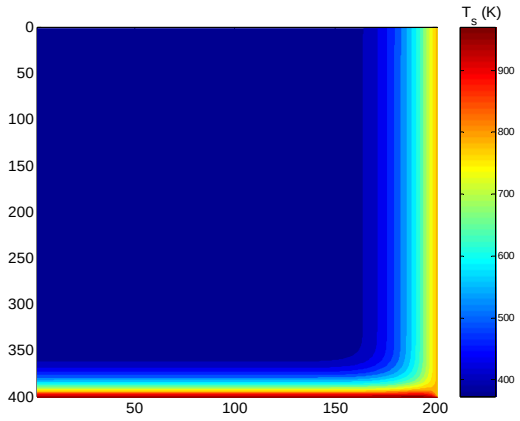


Figure 5: Solid temperature field (t=1000s)

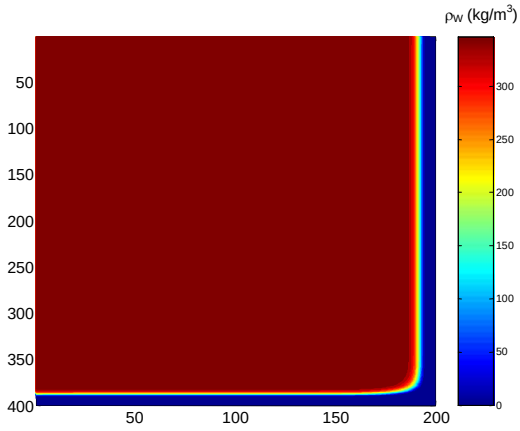


Figure 6: Wood apparent density field (t=1000s)

Preliminary sensibility tests

Even if the results obtained so far are not quantitatively accurate they allow a study of sensibility of the method to different parameters such as the grid discretisation or the time step.

The time step sensibility

The time step influence is very important, especially when solving analytically the equation (1). The most important differences appear on the reaction advancement front, visible in figure 6. The tests made in the same conditions and shown in figures 7 and 8 sustain this observation. Furthermore, it seems that larger the distance between the nodes produces an increased sensibility to the time step (results not shown here).

Figure 7 represents three temperature profiles calculated for a time step between 0.1 and 1 seconds, at 390mm from the top of the reactor.

This zone is the most sensible due to the fact that here the gradient of the apparent wood density is highest. The profiles in figure 8 at 200mm confirm the previous conclusion, this time the differences appearing on the vertical reaction front.

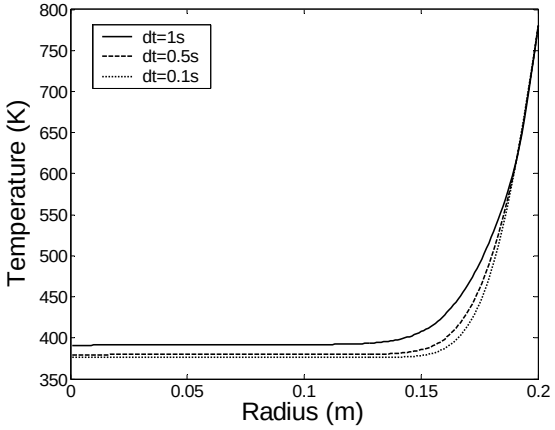


Figure 7: Solid temperature profiles at 390mm from the top (t=1000s) (400x200 nodes)

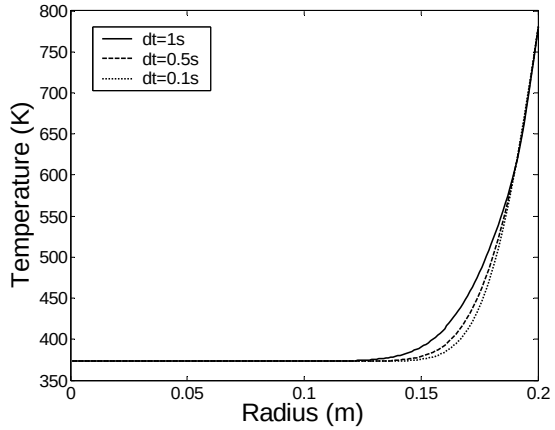


Figure 8: Solid temperature profiles at 200mm from the top (t=1000s) (400x200 nodes)

The same effect can be observed for the chemical data. In this case however, the differences are bigger due to a high sensibility of the kinetic constants to the temperature variations. Figure 9 presents the profiles for the wood apparent density on the reaction front at 390mm from the top of the reactor for the three considered time steps. In figure 10 at 380mm from the top, where the temperature is lower and the radial reaction front has not reached yet, the more noticeable differences are on the axial reaction front, close to the wall.

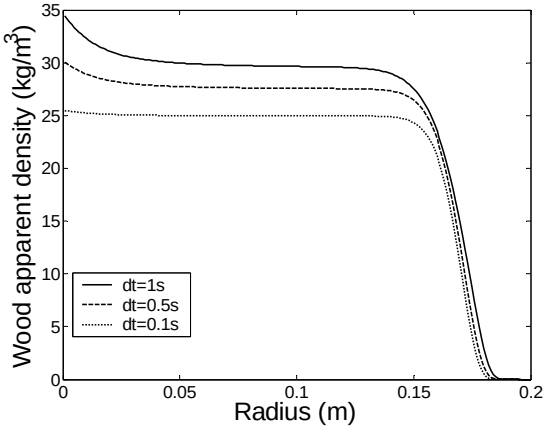


Figure 9: Wood apparent density profiles at 390mm from the top (t=1000s) (400x200 nodes)

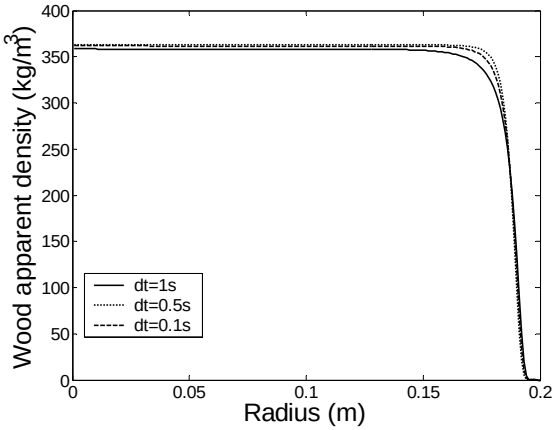


Figure 10: Wood apparent density profiles at 380mm from the top (t=1000s) (400x200 nodes)

The meshing sensibility

The test for the grid spacing sensibility are made for the same conditions as previous tests, for a time step of 0.1s. The retained data is presented in figures 11, 12 and 13.

Figures 11 and 12 compare the temperature profiles obtained for the first layers of the bed at 20mm, 10mm, 8mm and 5mm grid spacing with the corresponding profiles at 20mm, 10mm, 8mm and 5mm from the bottom of the reactor, calculated with a 1mm grid spacing. The results show that for a discretisation of less than 10mm, the temperature profiles are acceptable.

The chemical data seems to be more sensible at the grid spacing. As the detail in figure 13 shows, the wood apparent density profiles at 200mm from the top of the reactor has a pronounced sensibility zone on the vicinity of the axial reaction front, close to the wall. The resolutions obtained for 20mm, 10mm and 5mm grid spacing are far from the

values obtained for a 1mm discretisation, thus, from a chemical point of view, a grid spacing above the 1mm limit is highly inaccurate and inacceptable.

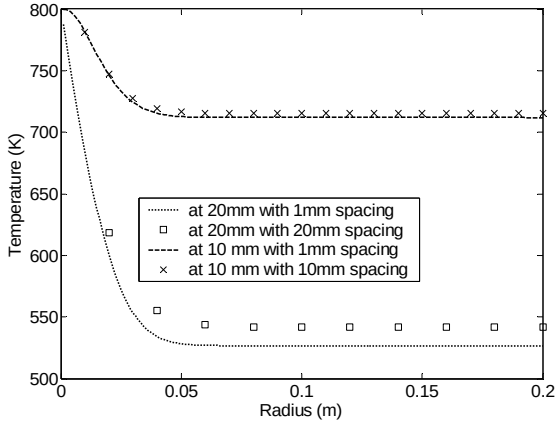


Figure 11: Temperature profile comparison at 10mm and 20mm from the bottom (t=1000s, dt=0.1s)

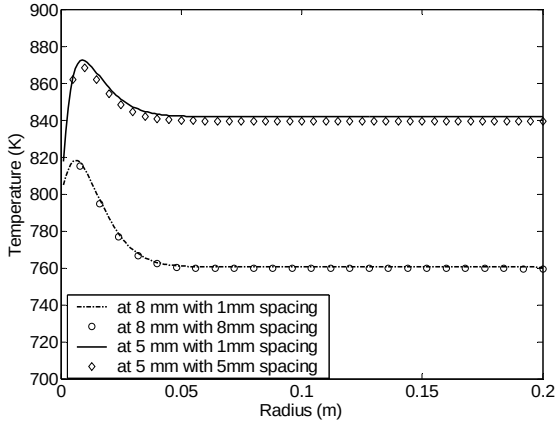


Figure 12: Temperature profile comparison at 8mm and 5mm from the bottom (t=1000s, dt=0.1s)

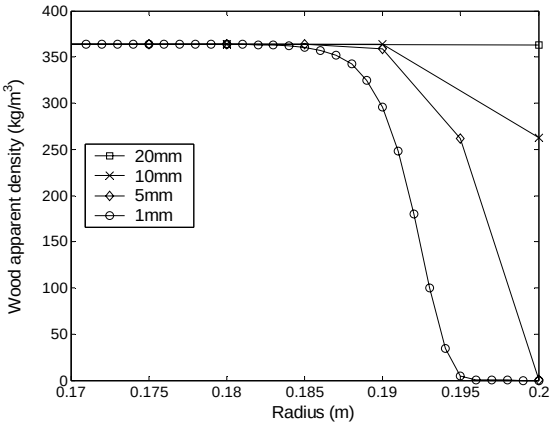


Figure 13: Chemical data for the discretization sensibility (t=1000s, dt=0.1s)

Conclusion

This work presents a nodal model of the pyrolysis process, as a link between a heat transfer model, a chemical model and a mass transfer model. The concepts of the method are shown and preliminary results are used to describe the sensibilities of the method to the spatial discretisation and to the imposed time step.

As a perspective, the mass transfer model can be implemented with the global model to allow the quantification of gas flow effects on the overall pyrolysis process.

References

[1] Saulnier J.-B., Alexandre A. *La modélisation thermique par la méthode nodale, ses succès et ses limites*. Revue Generale de Thermique 1985; 280 : 363 - 372

[2] Lagnotte P., Bertin Y., Saulnier J.-B. *Analyse de la qualité de modèles nodaux réduits à l'aide de la méthode des quadripôles*. International Journal of Thermal Sciences 1999; 38 : 51 – 65

[3] Belletre J., Sartre V., Biaist F., Lallemand A. *Transient state study of electric motor heating and phase change solid-liquid cooling*. Applied Thermal Engineering 1997; 17: 17-31

[4] Vijeu R. A., Tazerout M. *Modélisation et dimensionnement d'une zone de pyrolyse d'un gazogène co-courant.* In: Proceedings of COFRET'04, Nancy, 2004

[5] Patankar S.V. *Numerical heat transfer and fluid flow* - USA : Taylor & Francis, 1980

[6] Chen G., Andries J., Spliethoff H, Fang M. *Kinetics of biomass wastes pyrolysis for fuel gas production.* In: Proceedings of ECOS 2002

[7] Chen Y., Charpenay S., Jensen A., Wojtowicz M. A., Serio M. A. *Modeling of biomass pyrolysis kinetics*. In: Proceedings of Twenty-Seventh Symposium (International) on Combustion

[8] Babu B. V., Chaurasia A. S. *Pyrolysis of biomass: improved models for simultaneous kinetics and transport of heat, mass and momentum* Energy Conversion and Managemnt 2004; 45: 1297-1327

[9] Mousques M. *Modélisation du couplage réactions chimiques – transferts de chaleur/matière en vue du dimensionnement des réacteurs de pyrolyse* Ph. D. report, Université de Perpignan, 2001

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

CFD Biomass Two-stage Downdraft Gasifier Modelling: Oxidation-Reduction Zone

Luc Gerun*, Rãzvan Vîjeu, Jérôme Bellettre and Mohand Tazerout

Ecole des Mines de Nantes
Department of Energy and Environment

Abstract

The two-stage downdraft gasifier process is numerically analysed in this paper. The developed CFD model is divided into two parts, the oxidation and reduction zones, whereas the pyrolysis step is taken from literature. Chemical reactions are treated either by the "EDC" model (homogeneous reactions) and either by the Multiple Surface Reactions model (heterogeneous reactions). The solid phase is simulated by a Lagrangian particles tracking.
Although validation remains not completed, the model is in relatively good agreement with the trends taken from the literature.

Nomenclature

A: Surface area (m²)
C: Concentration (kmol. m^{-3})
C_2: Inertial resistance factor (m^{-1})
Cp: Heat capacity at constant pressure (J. kg^{-1}. K^{-1})
d: Diameter (m)
$D_{0,r}$: Molar diffusion coefficient (kmol. m^{-3})
E_a: Activation energy (J. $kmol^{-1}$)
h: Heat transfer coefficient (W. m^{-2}. K^{-1})
h^0: Standard state enthalpy (W. m^{-1}. K^{-1})
H: Heat of reaction (J. kg^{-1})
I: Radiative intensity (W)
k: Thermal conductivity (W. m^{-2}. K^{-1})
k_s: Frequency factor (s^{-1})
k_c : Arrhenius kinetic rate
k_m: Mass diffusion coefficient ($kg.m^{-2}.s^{-1}$)
m: Mass (kg)
M: Molar weight (kg. $kmol^{-1}$)
n: Index of refraction
p: Relative pressure (Pa)
R: Ideal gas constant (8 314 J. $kmol^{-1}$)
$R_{i,r}$: Homogeneous reaction rate (kg. $m^{-3}.s^{-1}$)
$R_{j,r}$: Rate of particle surface species depletion per unit area (kg. $m^{-2}.s^{-1}$)
$\overline{R_{j,r}}$: Rate of particle surface species depletion (kg. s^{-1})
T: Temperature (K)
Y: Mass fraction, $Y_i = \frac{m_i}{m_{total}}$

Greek Letters

α: Absorption coefficient (m^{-1})
α_p: Permeability (m^{-2}).
ε: Porosity, $\varepsilon = 1 - \frac{\rho_{apparent}}{\rho_{real}} = \frac{V_{void}}{V_{total}}$
ε_m: Emissivity
η_r: Effectiveness factor
λ: Thermal diffusivity (m^2. s^{-1})
ν: Kinematic viscosity (m^2. s^{-1})
ρ: Density (kg. m^{-3})
σ_S: Scattering coefficient (m^{-1})
σ: Boltzman constant ($5{,}67.10^{-8}$ $W.m^{-2}.K^{-4}$)
ϕ_T: Thiele modulus
ψ: Stream function

Subscript

eff: effective
g: gas
p: particle
r: reaction
s: solid

* corresponding author: e-mail: luc.gerun@emn.fr, phone number: (+33) 2.51.85.82.96, fax (+33) 2.51.85.82.99

1 Introduction

With the increasing concern for the environmental problems linked to the power production systems, the use of biomass as an alternative power source to the fossil fuels has become more attractive for the investors. In the perspectives of the Kyoto agreements, the reduction of CO_2 emissions gives to the energy production from biomass a considerable advantage.

Furthermore, if so far furnaces were the preferred way of energy production, the gasification with gas engine-generator technology is beginning to attract the interest not only of the scientific community, but also the attention of the industry due to its efficiency, its environment respect and the opportunity of producing both thermal and electrical energy [1].

Biomass gasification is a thermal conversion technology where a solid fuel is converted into a fuel gas. A partial combustion by a limited supply of oxidant, usually air, releases the heat necessary to the endothermic reactions of pyrolysis and reduction. The low LHV product gas is mainly composed by carbon monoxide, carbon dioxide, hydrogen, methane, water, nitrogen and various contaminants, such as small char particles, ash and tars.

Biomass is a particular kind of fuel. It is indeed often considered as a waste and its price is fixed by the collecting expenses. That is why small-scale installations are well adapted to this market. They allow valorising wastes directly on their production location.

In this context biomass downdraft gasifiers present indisputable advantages [1]. A traditional example is presented in *Figure 1* [2]. In further models called two-stage, pyrolysis and combustion/reduction zones are physically separated. This low tar level technology only requires a small gas cleaning unit to prevent engine fouling problems [3]. Furthermore it is well adapted to turnover and biomass diversity. Finally its utilisation is relatively easy and can be highly automated.

The technology is experimentally well known as such gasifiers have been used for several decades. However available kinetics data and the theoretical comprehension of the physical phenomena remain still insufficient to build a general accurate model.

Indeed complex chemical reactions occur in the reactor. They are moreover significantly dependent of temperature. But the turbulent two-phase flow and the geometry pattern make difficult to model accurately the heat transfer. That is why detailed modelling is required to determine the internal processes of oxidation and reduction.

In these perspectives, the objective of this study is to build a CFD model of the combustion-reduction zone. This tool will then be used to improve the knowledge of the physical phenomenon and to improve downdraft gasifiers design. Installation scale-up and tar level reduction are the two main aimed applications.

This model is coupled to a heterogeneous equilibrium based pyrolysis sub-model previously developed by Vîjeu and Tazerout [4]. As described in *Figure 2*, they form a global downdraft gasifier model. This tool can thus be used to design the whole reactor.

This paper describes the gasifier modelling. Firstly general equations and simulations tools are explained. The specificities of each submodel are then developed. Finally results of simulation are discussed.

2 Overall modelling principles

Gasifiers are complex reactors in which two-phase reactive turbulent flow occurs. CFD tool is used in order to simulate it. *Figure 2* shows the geometry of the simulated gasifier.

A segregated solver solves sequentially the continuity, momentum (Navier-Stokes equations), energy, radiation and species equations (N-1 equations for N species).

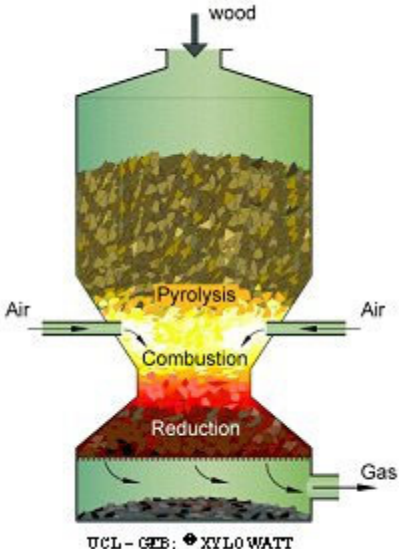


Figure 1: Traditional downdraft gasifier

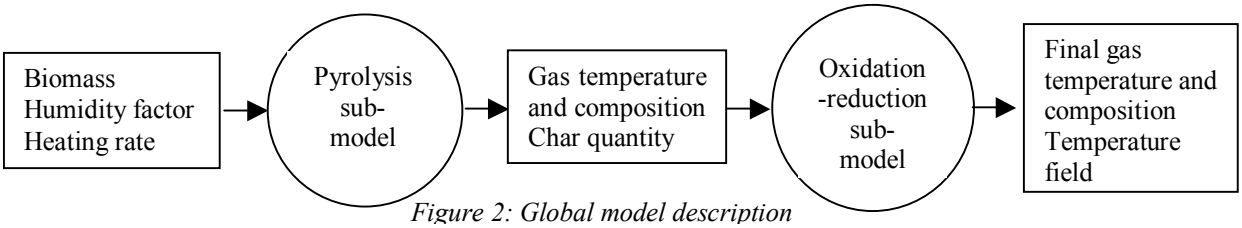


Figure 2: Global model description

Governing equations are converted to algebraic equations that can be solved numerically by a control-volume-based technique through a power-law scheme. SIMPLE algorithm couples velocity and pressure.

The 2D axisymmetric unstructured grid is composed of 10,000 cells. Meshing tests have been performed from 5,000 to 24,000 cells to insure independence of results.

Simulation is considered as converged when residuals remain constant at a value below 10^{-4} (10^{-2} for momentum). The calculation time is approximately fifty hours on a biprocessor 1.2 GHz computer, when both oxidation and reduction submodels are running in parallel.

The model is indeed split into two submodels, one for the oxidation zone, the other for the fixed bed where reduction reactions occur, as shown in *Figure 3*. The input of the oxidation part is taken from the pyrolysis model.

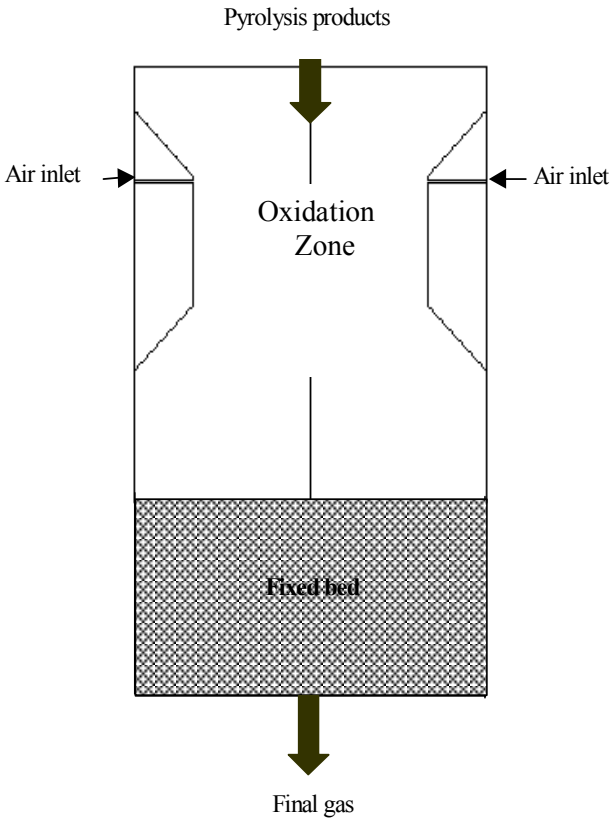


Figure 3: Gasifier model geometry

3 Oxidation zone

Reynolds number is around 5,000 at the inlet where turbulence rate is 1%. The flow is thus turbulent because of the reactor geometry. Moreover velocity pattern is complex as recirculation may occur. A detailed turbulence model is thus required to simulate these phenomenon. After a sensitivity study, the RNG k-ε model was chosen as the best compromise between accuracy and efficiency.

Turbulence is also taken into account in the chemical reaction model, "Eddy Dissipation Concept" model [5] that calculates Arrhenius rate at the turbulence time scale. It seems to suit well to this kind of reactive flow [6].

As temperature exceeds 1000°C in the reactor core, heat transfer by radiation is predominant. They are simulated by Discrete Ordinates model [7], a four-flux method. The gas emissivity is computed by the cell-based Weighted-Sum-of-Gray-Gases model that takes into account the concentration of CO_2 and H_2O.

Model equations are listed in *Table 1*.

Wall boundary conditions are simulated by a low Reynolds number approach [8].

The chemical mechanism consists in three oxidation reactions (R1-3), the water-gas shift reaction (R4) and the methane reformation (R5).

$$CO + 0.5\ O_2 \longrightarrow CO_2 \quad \text{(R1)}$$

$$CH_4 + 1.5\ O_2 \longrightarrow CO + 2\,H_2O \quad \text{(R2)}$$

$$H_2 + 0.5\,O_2 \longrightarrow H_2O\ (g) \quad \text{(R3)}$$

$$CO + H_2O \longrightarrow CO_2 + H_2 \quad \text{(R4)}$$

$$CH_4 + H_2O \longrightarrow CO + 3\,H_2 \quad \text{(R5)}$$

Arrhenius rate is calculated from equation (1).

$$k_c = k_s e^{-\frac{E_a}{RT}} [A]^a [B]^b \quad (1)$$

Constants for are taken from literature. They are listed in *Table 2*.
Solid phase is simulated by a Lagrangian particles tracking. It is observed that particles residence time is relatively short (<1s) in a two-stage downdraft gasifier. As a consequence heterogeneous reactions are negligible, as oxygen reacts far quicker with gas phase.
Heat transfer from fluid to solid phase is calculated by equation (2).
Final particles temperature is then used to set initial conditions in the bed.

$$m_p Cp_p \frac{dT_p}{dt} = hA_p\left(T_\infty - T_p\right) - \frac{dm_p}{dt} H_r + A_p \varepsilon_p \sigma (T_\infty^4 - T_p^4) \quad (2)$$

4 Reduction zone: the fixed bed

The gasifier is considered working on a stable and steady state, that is that the mass of char depleted by reduction reactions is equal to the mass of char coming from the oxidation zone.
Particle mass flowrate is assumed to be uniform over the surface and constant. The bed of average density 250 kg/m^3 consists in 90 colons of ten-millimetre-diameter particles. They are injected at the bed top and put off the calculation at the bottom (ash losses).

4.1 Particles model

Even if particles velocities are almost null, particles are modelled as a discrete phase in a Lagrangian frame of reference by the Discrete Phase model. The trajectory calculation takes only into account interactions with walls, as the grate compensate gravity and drag forces. As a result particles velocity gradient is almost null.
Initial velocity is determined by the char mass flowrate from pyrolysis model. Their temperature is calculated by equation (2). Convection, radiation and heat of reactions are included. However particles are considered isolated, as conduction and collisions are neglected, being smaller by an order of 20 compared to radiative heat transfer.

4.2 Pressure drop

A source term of friction from Ergun equation (9) due to the porosity of the bed is included in the momentum equation.

$$S_i = -\left(\frac{\mu}{\alpha_p} v_i + C_2 \frac{1}{2} \rho |v_i| v_i\right) \quad (3)$$

with $\alpha_p = 3*10^9$ m^{-2} and $C_2 = 3.4*10^7$ m^{-1}.

4.3 Surface reactions

Only two surface reactions (R6) and (R7) are considered. Indeed oxygen has been totally consumed in the oxidation zone whereas C-H_2 reaction is far slower than C-CO_2 and C-H_2O and thus is negligible [9].

$$C + H_2O \longrightarrow CO + H_2 \quad (R6)$$

$$C + CO_2 \longrightarrow 2\,CO \quad (R7)$$

Arrhenius rate constants are deduced from Groeneveld and Van Swaaij [10] (see *Table 2*).
Reaction rate are calculated by the Multiple Surface Reactions model, based on the works of Smith [11].
From mass-balance considerations the particle depletion rate can be expressed as:

$$R = D_0(C_g - C_s) = k_c (C_s)^N \quad (4)$$

Implemented for each reaction r and each species j, it gives a set of two equations:

$$\overline{R}_{j,r} = A_p \eta_r Y_j R_{i,r} \quad (5)$$

$$R_{j,r} = R_{chem}\left(p_n - \frac{R_{j,r}}{D_{o,r}}\right)^{N_r} \quad (6)$$

with $D_{o,r} = C_{1,r} \dfrac{\left(\dfrac{T_p + T_\infty}{2}\right)^{0,75}}{d_p}$;

$C_{1,r}$ is a constant calculated from molar and Knudsen diffusion rate coefficient for the mean temperature of the bed top; $C_{1,r} = 4.968*10^{-10}$ for CO_2 and $7.769*10^{-10}$ for H_2O;

and $\eta_r = \dfrac{3}{\phi_T}\left(\dfrac{1}{\tanh \phi_T} - \dfrac{1}{\phi_T}\right)$. (7)

This set of equations is solved by iteration except if N_r is equal to 0 or 1. For the last case the analytical solution is:

$$\overline{R}_{j,r} = A_p \eta_r Y_j p_n \frac{R_c D_{0,r}}{D_{0,r} + R_c} \quad (8)$$

The particles diameter remains constant by simplification while their density is allowed to change. Moreover inhibition and degree of conversion are not taken into account.
Oxidation reactions are neglected in the bed as oxygen concentration is closed to zero.

Table 1: Governing equations

Mass $$\frac{\partial \rho}{\partial t} + \nabla.(\rho \vec{v}) = S_m \quad (E-1)$$

Momentum $$\frac{\partial}{\partial t}(\rho \vec{v}) + \nabla.(\rho \vec{v}\vec{v}) = -\nabla p + \nabla.(\bar{\bar{\tau}}) + \rho \vec{g} + \vec{F} \quad (E-2)$$

Energy $$\frac{\partial}{\partial t}(\rho E) + \nabla.(\vec{v}(\rho E + p)) = \nabla.\left[k_{eff} \nabla T - (\sum_j h_h \vec{J}_j) + (\bar{\bar{\tau}}.\vec{v}) \right] + S_f^h \quad (E-3)$$

with $$k_{eff} = \lambda \nu_{eff} Cp$$

Radiative transfer $$\nabla.(I(\vec{r},\vec{s})\vec{s}) + (\alpha + \sigma_S) I(\vec{r},\vec{s}) = \alpha n^2 \frac{\sigma T^4}{\pi} + E_p + \frac{\sigma_S}{4\pi} \int_0^{4\pi} I(\vec{r},\vec{s}) \Phi(\vec{s}.\vec{s}') d\Omega' \quad (E-4)$$

Turbulence $$\frac{\partial \left(\rho U_j k \right)}{\partial x_j} = \frac{\partial}{\partial x_j} \left[\rho \alpha_k \nu_{eff} \frac{\partial k}{\partial x_j} \right] + \rho \nu_t \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \frac{\partial U_i}{\partial x_j} - \rho \varepsilon \quad (E-5)$$

$$\frac{\partial \left(\rho U_j \varepsilon \right)}{\partial x_j} = \frac{\partial}{\partial x_j} \left(\rho \alpha_\varepsilon \nu_{eff} \frac{\partial \varepsilon}{\partial x_j} \right) + C_{\varepsilon 1} \frac{\varepsilon}{k} \rho \nu_t \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \frac{\partial U_i}{\partial x_j} - C_{\varepsilon 2} \rho \frac{\varepsilon^2}{k} - R \quad (E-6)$$

$$\text{with } \nu_{eff} = \nu \left(1 + \sqrt{\frac{C_\mu}{\nu}} \frac{k}{\sqrt{\varepsilon}} \right)^2 = \nu \left(1 + \sqrt{\nu_t / \nu} \right)^2$$

$C_\mu = 0{,}0845$; $C_{\varepsilon 1} = 1{,}42$ and $C_{\varepsilon 2} = 1{,}68$. α_k and α_ε are the inverse of turbulent Prandtl number for k and ε.

Species transport $$\frac{\partial}{\partial t}(\rho Y_i) + \nabla.(\rho \vec{v} Y_i) = -\nabla.\left(-\left(\rho D_{0,i} + \frac{\mu_t}{Sc_t} \right) \nabla Y_i \right) + R_i \quad (E-7)$$

Homogeneous reaction rate $$R_{i,r} = \frac{(\xi^*)^2 \chi}{\tau^*} (Y_i^* - Y_i) \quad (E-8)$$

With $\xi^* = C_\xi \left(\frac{\upsilon \varepsilon}{k^2} \right)^{\frac{3}{4}}$; $\tau^* = C_\tau \left(\frac{\upsilon}{\varepsilon} \right)^{\frac{1}{2}}$; $\chi = \frac{\rho}{[1 - (\xi^*)^3]}$; $C_\xi = 2{,}1377$; $C_\tau = 0{,}4082$.

Table 2: Reference value for kinetic constants

Reaction	k_s (s^{-1})	E_a (J. $kmol^{-1}$)	A	B	Reference
(R1)	$2.2*10^{12}$	$1.67*10^8$	1	0.5	Bettahar [12]
(R2)	$5.012*10^{11}$	$2*10^8$	0.7	0.8	Fluent [13]
(R3)	$4.462*10^{12}$	$4.2*10^7$	1	1	Di Blasi [9]
(R4)	124.03	$1.26*10^7$	1	1	Di Blasi [9]
(R5)	312	$2.07*10^8$	1	0	Liu [14]
(R6)	26.25	$2.171*10^8$	n = 0.7		Groeneveld [10]
(R7)	26.25	$2.171*10^8$	n = 0.7		Groeneveld [10]

5 Results

The model has been applied to a representative downdraft gasifier of diameter 0.9 m. It is feed with 15%-humidity beech wood at a mass flowrate of 20 kg/h and a mass fuel – air ratio of 1:1 for a power of approximately 100 kW thermal. Air is preheated up to 670K before injection.

Pyrolysis is supposed to occur at 870 K. Pyrolysis products calculated from the model of Vijeu and Tazerout [4] are char, 30%, and gas, 70%. Its mass composition is approximately 50% CO_2, 25 % water, 25% CH_4 and traces of CO and H_2.

The temperature profile in the oxidation zone, shown in *Figure 4*, is conform to the expectations. A hot point is localised just at the exit of the air injection nozzle, where combustion occurs. Elsewhere temperature is relatively uniform in the radial direction, with moderate axial gradient.

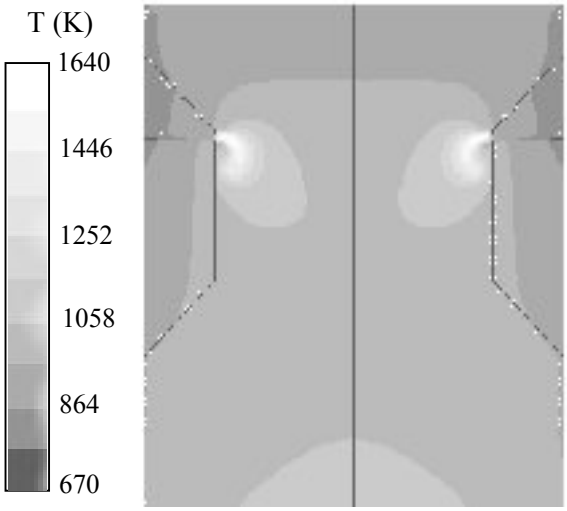


Figure 4: Temperature profile in the oxidation zone

A recirculation occurs in the back of the throat (see *Figure 5*). Velocity field show that 1D approximation is not correct for the gasifier type. Indeed radial velocity may not be neglected. It is noticeable that the flow is strongly influenced by biomass and air feed rate.

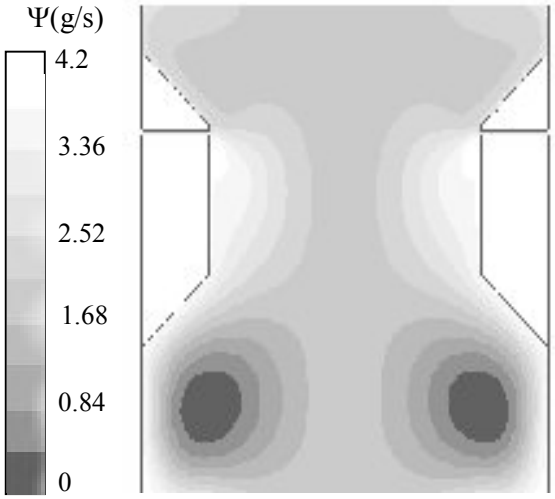


Figure 5: Velocity stream line

The conjugation of temperature and axial velocity (see *Figure 6*) knowledge is particularly interesting for gasifier design. It is indeed commonly admitted in the literature [15] that tar cracking depends of the temperature and of the residence time.
In this case the zone of temperature hotter than 1,100°C is very restricted, that supposes thus a relatively low tar cracking.

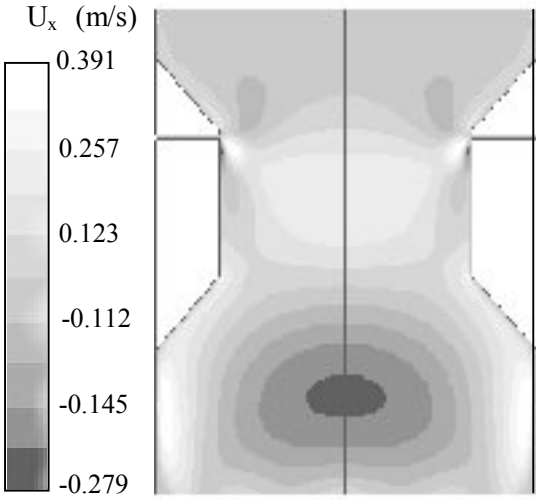


Figure 6: Axial velocity field

Oxygen evolution enlightens the zone where combustion occurs. *Figure 7* shows that oxygen is quickly consumed as kinetic rate are high.

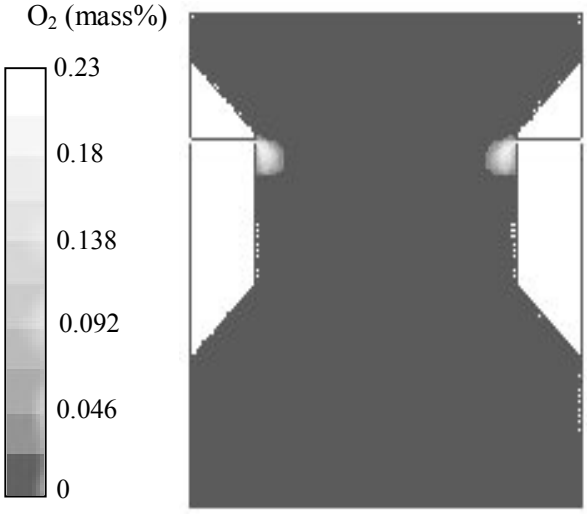


Figure 7: Oxygen mass concentration

Carbon dioxide is main species of the mixture. It is produced during combustion and also at a lower magnitude by the water-gas (R4) reaction. Its concentration decreases because of a the dilution by nitrogen as shown in *Figure 8*.
Final gas composition, given *Table 3*, corresponds to classical values for downdraft gasifiers [3, 16].

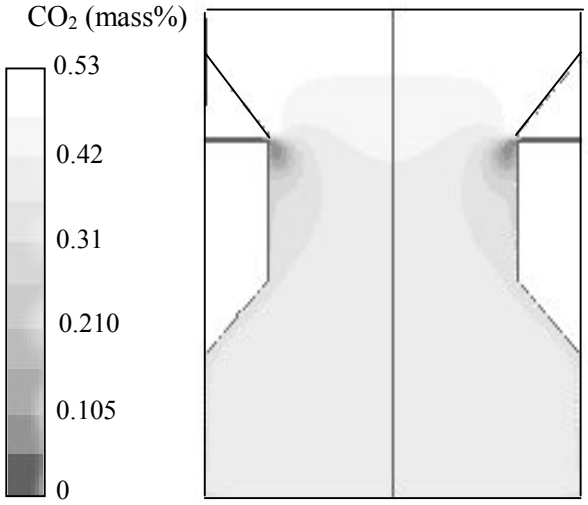


Figure 8: Carbon dioxide mass

Methane and carbon monoxide concentrations are however over-predicted. Tars are indeed not simulated and they are also assimilated mainly in these two species.

Table 3: Final gas composition

	CO	CO_2	H_2	CH_4	N_2
Volume % (dry basis)	22.5	14.6	24.7	5.6	32.6

6 Conclusion

A downdraft gasifier CFD 2D model has been built. Although its validation is only partial, particularly in the packed bed, this tool predicts detailed process comportment. It takes into account turbulence, surface reactions, heat transfer with radiation included.
Results show that the model can be a useful tool for designing gasifier. It must however be still fully validated to guarantee the accuracy of data. Besides the reduction zone model needs to be improved so that catalyst effect of ashes, hydrogen inhibition and conversion degree would be taken into account.

References

[1] McKendry, P., *Energy production from biomass (part 3): gasification technologies*, Bioresource Technology 2002;83;55-63.

[2] Tchouate Heteu, P. M., *Feasibility of biomass gasification for decentralised rural electrification and industry in remote areas in Africa*, In: Proceedings of the First international Conference and Exhibition on Biomass for energy and industry, Seville, 1999.

[3] Henriksen, U., Ahrenfeldt, J., Gøbel, B., Hindsgaul, C. and Jensen, T. K., *The Viking gasifier*, In: Proceedings of the IEA-meeting, London, 2003.

[4] Vijeu, R. and Tazerout, M., *Modélisation et dimensionnement d'une zone de pyrolyse d'un gazogène co-courant*, In: Proceedings of COFRET'04, Nancy, 2004.

[5] Magnussen, B. F., *On the structure of turbulence and a generalised eddy dissipation concept for chemical reaction in turbulent flow*, In: Proceedings of the Nineteenth AIAA Meeting, St Louis, 1981.

[6] Bellettre, J. and Tazerout M., *Numerical simulation of reactive flows in a wood gasifier*, IN: Proceedings of the Third Mediterranean Combustion Congress, Marrakech, 2003.

[7] Chui, E. H. and Raithby, G. D., *Computation of radiant heat transfer on a non-orthogonal mesh using the finite-volume method*, Numerical Heat Transfer 1993;23;269-288.

[8] Lopez-Matencio J. L., Bellettre J. and Lallemand A., *Numerical prediction of turbulent heat and mass transfer above a porous wall subjected to vaporisation cooling*, I. J. Transport Phenomena, 2003;5(3);185-201.

[9] Di Blasi, C., *Dynamic behaviour of stratified downdraft gasifiers*, Chemical Engineering Science 2000;55;2931-2944.

[10] Groeneveld, M. J. and Van Swaaij, W. P. M., *Gasification of char particles with CO2 and H2O*, Chemical Engineering Science. 1980;35;307-313.

[11] Smith, I. W., *The combustion rates of coal chars: a review*, In: Proceedings of the 19th Symposium (International) on Combustion/The Combustion Institute, Haifa, 1982.

[12] Bettahar, A., *Modélisation et Simulation Numérique de la Gazéification du Bois*, Master thesis, Nantes, 2002.

[13] Fluent 6, *User's Guide*, Lebanon: Fluent Inc., 2001.

[14] Liu, X. J., Zhang, W. R., et Park, T. J., *Modelling coal gasification in an entrained flow gasifier*, Combustion Theory and Modelling 2001;5;595-608.

[15] Milne, T. Abatzoglou N. and Evans, R. J., *Biomass Gasifier "TARS": Their nature, formation and conversion*, report NREL/TP-570-25357, IEA Bioenergy Biomass Utilisation, 1998.

[16] Reed, T. and Das, A., *Handbook of biomass gasifier engine systems*, 2nd edition, Golden: Biomass Energy Foundation (BEF) Press, 1988.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

DYNAMIC SIMULATION OF A SOLAR COLLECTOR FIELD WITH INTELLIGENT DISTRIBUTED PARAMETER MODELS

Esko K. Juuso*
Control Engineering Laboratory
Department of Process and Environmental Engineering
FIN-90014 University of Oulu, Finland

Abstract

The dynamic models of the solar collector field provide a smooth and accurate overall behaviour achieved with linguistic equations combined special situation handling with fuzzy systems. The new adaptive control technique tuned with this simulator has reduced considerably temperature differences between collector loops. Efficient energy collection was achieved even in variable operating condition. The new distributed parameter model is also aimed for control design. It extends the operability of the simulator to evaluating the controller performance for drastic changes, e.g. startup and large load disturbances, and local disturbances and malfunctioning.
Keywords: Solar power plant, dynamic modelling, distributed parameter systems, nonlinear models, linguistic equations, fuzzy set systems

Introduction

Solar power varies independently and cannot be adjusted to suit the desired demand. The aim of solar thermal power plants is to provide thermal energy for use in an industrial process such as seawater desalination or electricity generation. To provide a viable, cost effective alternative for power production, they must achieve this task despite fluctuations in their primary energy source, the sunlight. In addition to seasonal and daily cyclic variations, the intensity depends also on atmospheric conditions such as cloud cover, humidity, and air transparency.

Any available thermal energy is collected in a usable form, i.e. at the desired temperature range, which improves the overall system efficiency and reduces the demands placed on auxiliary equipment as the storage tank. In cloudy conditions, the solar field is maintained in a state of readiness for the resumption of full-scale operation when the intensity of the sunlight rises once again. Unnecessary shutdowns and start-ups of the collector field are both wasteful and time consuming [1, 2].

The control scheme should regulate the outlet temperature in order to supply energy as constant as possible despite the disturbances and uncertainties in solar radiation and energy demand [1, 3, 4]. An overview of possible control strategies presented in [5] include basic feedforward and PID schemes, adaptive control, model-based predictive control, frequency domain and robust optimal control and fuzzy logic control. A comparison of different intelligent controllers is presented in [6]. A linguistic equation (LE) controller was first implemented on a solar collectors field [1, 2]. Later adaptive set point procedure and feed forward features have been included for avoiding overheating. The present controller takes also care of the actual set points of the temperature [7].

Trial and error type controller tuning does not work since the operating conditions cannot be reproduced since the dynamics depends on the operating conditions. Models have been integrated to various control schemes. Feedforward approaches based directly on the steady state energy balance relationships can be based on measurements of solar radiation and inlet temperature [8]. A feedforward controller has been combined with different feedback controllers, even PID controllers operate for this purpose [9]. A model-based predictive control is another possibility to take into account nonlineari-

*Corresponding author. Phone: +358-8-5532463, Fax: +358-8-5532304, E-mail: esko.juuso@oulu.fi

ties [10]. The classical internal model control (IMC) can operate efficiently in varying time delay conditions [11]. The adaptation scheme of LE controllers is extended by a model-based handling of the operating conditions [7].

Dynamic simulators are needed in controller design and tuning. Also process optimisation could be improved by modelling and simulation. Lumped parameter models taking into account the sun position, the field geometry, the mirror reflectivity, the solar radiation and the inlet oil temperature have been developed for a solar collector field [5]. Dynamic simulators based on linguistic equations are continuously used in development of multilayer linguistic equation controllers. The robust dynamic simulator based on linguistic equations is an essential tool in fine–tuning of these controllers [12, 13].

In this paper, the data-driven LE modelling approach is combined with a dynamic energy balance and extended to developing distributed parameter models based on process measurements obtained in several test campaigns at a solar plant.

Solar power plant

All the experiments were carried out in the *Acurex Solar Collectors Field of the Plataforma Solar de Almeria* located in the desert of Tabernas (Almeria), in the south of Spain. The *Acurex field* supply thermal energy (1 MW) in form of hot oil to an electricity generation system or a Multi–Effect Desalination Plant. The solar field consists of twenty rows of east-west oriented, one-axis elevation-tracking, parabolic-trough collectors (Figure 1) in ten parallel loops (Figure 2). There are 480 modules in the field. The total reflective aperture area of the ACUREX collector field is 2,674 m^2. The hot oil goes to a 5 MWh_t thermal storage system consisting of a 140 m^3 thermocline oil tank with automatic fire-extinguishing and volatile-condensing systems. Quick cooling can be done with a water-cooled oil cooler. A desalination plant is is connected to this storage system.

Control is achieved by means of varying the flow pumped through the pipes during the plant operation. In addition to this, the collector field status must be monitored to prevent potentially hazards situations, e.g. oil temperatures greater than 300 ^oC. When a dangerous condition is detected software automatically intervenes, warning the operator and defocusing the collector field. As the control is fast and well damped, the plant can be operated close to the design limits thereby improving the productivity of the plant.

The energy balance of the collector field can be represented by expression [9]

$$I_{eff}A_{eff} = (1-\eta_p)F\rho cT_{diff} \tag{1}$$

where I_{eff} is effective irradiation (Wm^{-2}), A_{eff} effective collector area (m^2), η_p a general loss factor, F flow rate of the oil (m^3s^{-1}), ρ oil density kgm^{-3}, c specific heat of oil $(Jkg^{-1}K^{-1})$ and T_{diff} temperature difference between the inlet and the outlet (^oC). The effective irradiation is the direct irradiation modified by taking into account the solar time, declination and azimuth. By combining the oil characteristics and geometrical parameters into a term

$$\alpha = \frac{A_{eff}}{(1-\eta_p)\rho c}, \tag{2}$$

a simple feedforward controller is obtained:

$$F = \alpha\frac{I_{eff}}{T_{out}-T_{in}}. \tag{3}$$

The temperature increase in the field may rise upto 110 degrees which means that the gain of the system is affected strongly by the variations of density, viscosity, and specific heat with the temperature (Figure 3). The daily operation is started with minimum flow, and the flow increase must be quite moderate during the whole startup phase since the high viscosity does not allow very high flow. In the beginning of the daily operation, the oil is circulated in the field, and the valve to storage system is open when an appropriate outlet temperature is achieved.

An extensive number of daily data sets is available from various test campaigns as the process must be controlled all the time. Test campaigns include step changes and load disturbances but they cannot be planned in detail because of changing weather conditions. Weather conditions take care of radiation disturbances.

Modelling is based on process data from controlled process characterised by time varying transport delays, oscillations and fast disturbances of solar irradiation. The model described in this paper have been developed from the data of the test campaigns of the

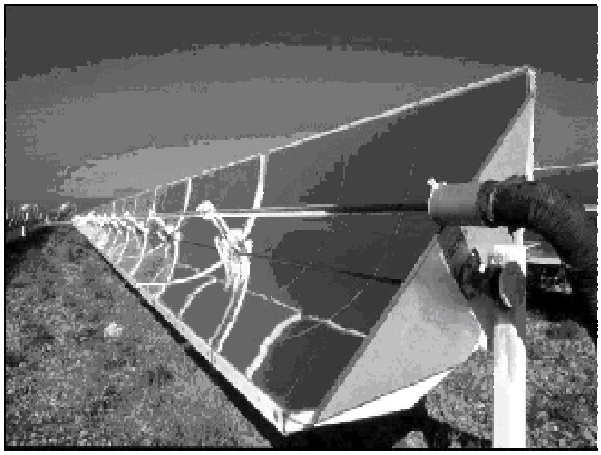


Figure 1: A receiver at the Acurex solar collector field.

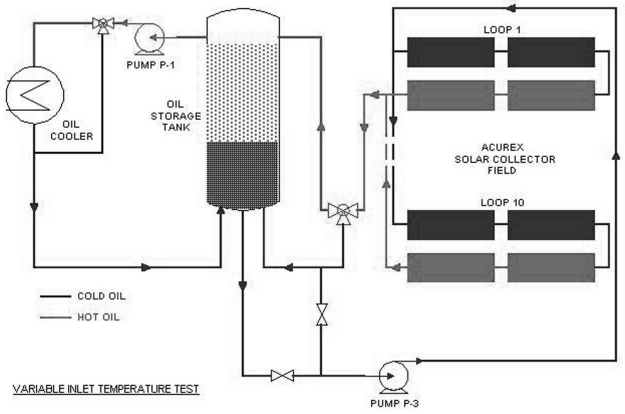


Figure 2: Layout of the Acurex solar collector field.

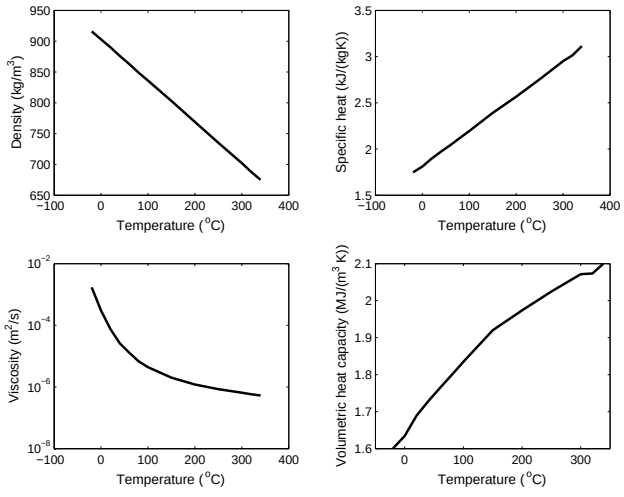


Figure 3: Oil properties: density, specific heat, viscosity and volumetric heat capacity, $C_V = \rho c$.

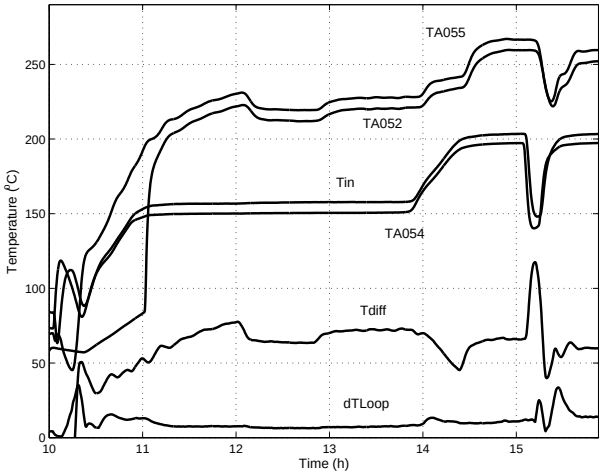


Figure 4: Temperatures and temperture differences in the field (June 21, 2002).

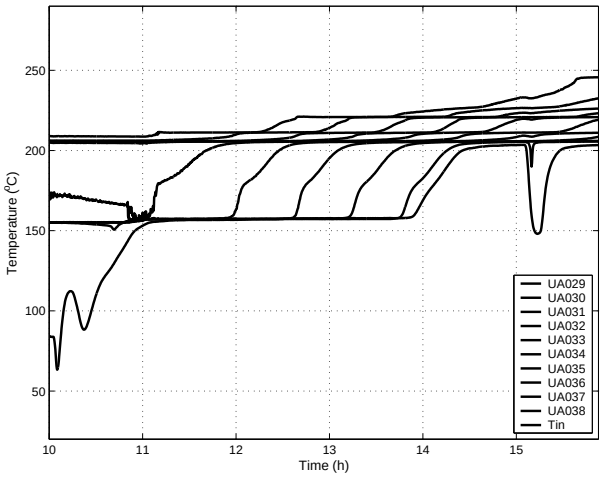


Figure 5: Temperatures on different levels of the thermocline (June 21, 2002).

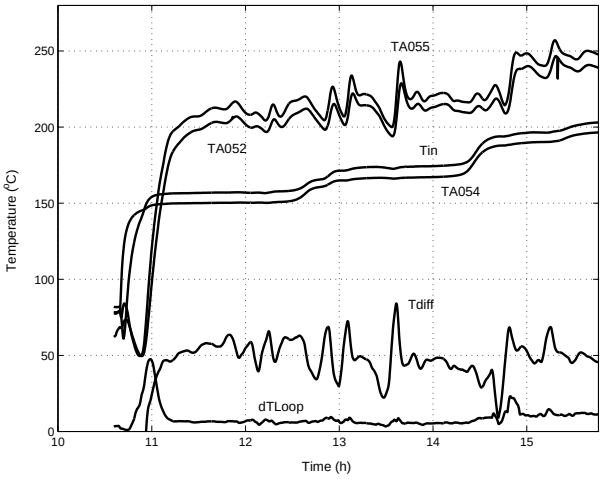


Figure 6: Temperature and temperature differences in the field in cloudy conditions (June 12, 2002).

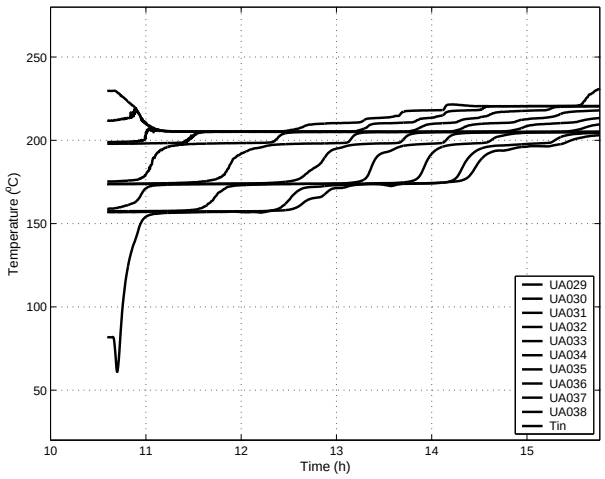


Figure 7: Temperatures on different levels of the thermocline in cloudy conditions (June 12, 2002).

LE controllers [7]. Figures 4 and 5 present temperatures in a test with fairly smoothly changing irradiation. In the end of the period, there is a slow increase of the inlet temperature followed with a very large load disturbance. The changes of the temperature difference are results of the setpoint tracking. The outlet oil goes to the top of the thermocline, and the temperatures in the lower levels of the tank increase gradually when the oil of the lower levels goes back to the field. On cloudy conditions, the control is much more difficult [7]. Figures 6 and 7 present temperatures on a cloudy day with fluctuating irradiation.

The collector loops do not operate identically: the difference between the highest and the lowest loop temperature, $dTLoop$, increases when the flow of the oil is changed rapidly. This happens in the startup phase and when load disturbances are introduced.

Dynamic modelling

For nonlinear multivariable modelling on the basis of data with understanding of the process there are two alternatives: fuzzy set systems and linguistic equations. In intelligent control design, hybrid techniques combining different modelling methods in a smooth and consistent way are essential for successful comparison of alternative control methods. Switching between different submodels in multiple model approaches should be as smooth as possible. For slow processes, predictive model–based techniques are necessary at least on the tuning phase. Adaptation to various nonlinear multivariable phenomena requires a highly robust technique for the modelling and simulation.

Linguistic equation models consist of two parts: *interactions* are handled with linear equations, and nonlinearities are taken into account by *membership definitions* [6]. The basic element is a compact equation

$$\sum_{j=1}^{m} A_{ij} X_j + B_i = 0, \qquad (4)$$

where X_j is a linguistic level for the variable j, $j = 1...m$. The direction of the interaction is represented by interaction coefficients A_{ij}. The bias term B_i was introduced for fault diagnosis systems. Linguistic equations can be used to any direction. The membership definition is a nonlinear mapping of the variable values inside its range to a certain linguistic range, usually $[-2,2]$. The mapping is represented with two monotonous, increasing functions, which must overlap in the center at the linguistic value 0. In the present system, these functions are second order polynomials. Coefficients are extracted from data or defined from expert knowledge. [14]

Dynamic intelligent models can be constructed on the basis of state–space models, input–output models or semi–mechanistic models. In the state–space models, fuzzy antecedent propositions are combined with a deterministic mathematical presentation of the consequent. The structure of the input–output model establishes a relation between the collection of past input–output data and the predicted output. Multiple input, multiple output (MIMO) systems can be built as a set of coupled multiple input, single output MISO models. Delays are taken into account by moving the values of input variables correspondingly.

The basic form of the *LE* model also is a static mapping, and therefore dynamic *LE* models could include several inputs and outputs originating from a single variable [6]. However, rather simple input-output models, e.g. the old value of the simulated variable and the current value of the control variable as inputs and the new value of the simulated variable as an output, can be used since nonlinearities are taken into account by membership definitions. Comparisons with different parametric models, e.g. autoregressive moving average (*ARMAX*), autoregressive with exogeneous inputs (*ARX*), *Box-Jenkins* and

Output-Error (*OE*), show that the performance improvement with additional values is negligible. [14].

Collector field models

Distributed parameter model can be based on the energy balance: energy stored = Irradiance - Energy transferred - Heat loss. For a unit volume this can be represented by

$$\rho cA\frac{\partial T}{\partial t} = I_{eff}W\eta_0 - \rho cF\frac{\partial T}{\partial x} - hD(T - T_{amb}) \quad (5)$$

where A is cross section of the pipe line (m^2), c specific heat of oil $(Jkg^{-1}K^{-1})$, D pipe diameter (m), I_{eff} irradiation (Wm^{-2}), h heat transfer coefficient $(Wm^{-2}K^{-1})$, T oil temperature (^oC), T_{amb} ambient temperature (^oC), x length coordinate (m), F flow rate (m^3s^{-1}), W width of the mirror (m), η_0 optical efficiency, ρ oil density kgm^{-3}, t time (s).

Conventional mechanistic models do not work: there are problems with oscillations and irradiation disturbances. Oil properties change drastically with temperature, and therefore operating conditions change considerably during the working day, e.g. during the startup stage, the oil flow is limited by the high viscosity.

Equation (5) can be represented by

$$\frac{T_i(t+\Delta t) - T_i(t)}{\Delta t} = a_1 I_{eff} + a_2 T_i(t) + a_3 T_{amb} \quad (6)$$

where coefficients a_1, a_2 and a_3 depend on operating conditions, and therfore, the process is highly nonlinear.

Location of the *ith* element depends on the flow rate:

$$\frac{x_i(t+\Delta t) - x_i(t)}{\Delta t} = \frac{F(t)}{V_{tot}}, \quad (7)$$

where V_{tot} is the total amount of the oil in the collector field. Actually, the volume of the oil depends on temperature.

Coefficients a_1, a_2 and a_3 depend on the operating conditions. As the linguistic equation models are based on nonlinear scaling of variables, the corresponding coefficients of the LE models are constant on a wide operating area.

Operating areas

The working point variables already define the overall normal behaviour of the solar collector field. The

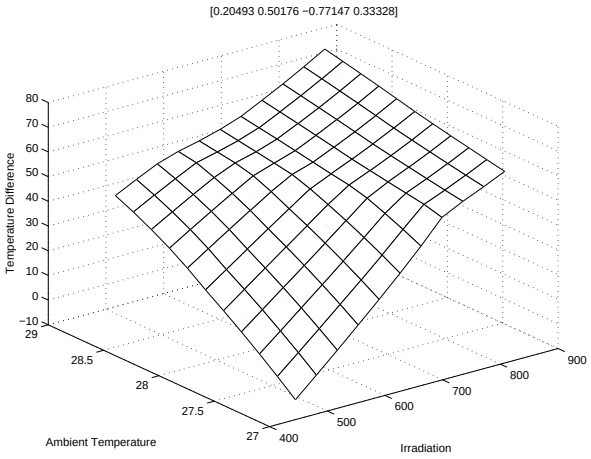


Figure 8: Model surface of a LE model for the working point variables [15].

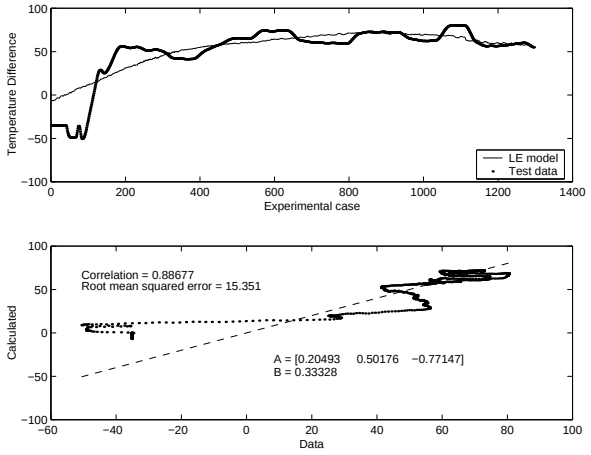


Figure 9: Simulation results for a LE model of the working point variables [15].

model of typical operation shown in Figure 8 has a quite high correlation to the real process data (Figure 9). The differences have a clear relation to operating conditions, e.g. oscillatory behaviour is a problem when the temperature difference is higher than the normal. The startup phase needs a special model. Separate dynamic models are needed to capture the dynamic behaviour in different operating conditions: the model surface of the normal model is presented in Figure 10. The model shown in Figure 8 corresponds closely to the model presented by Equation (6). The typical normal operation shown here is not always optimal.

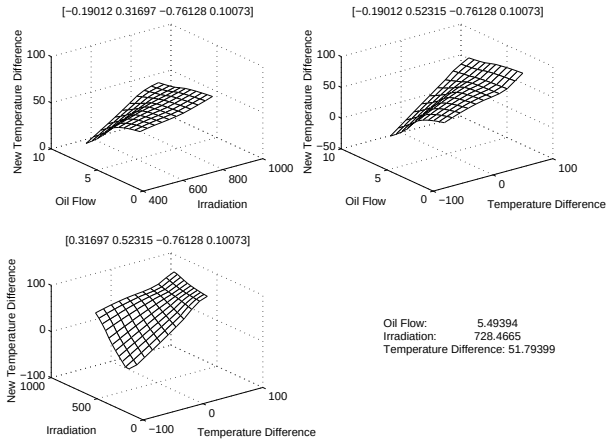


Figure 10: A dynamic LE model for temperature difference [15].

Lumped parameter models

Dynamic linguistic equation (LE) models provide a good overall behaviour in different operating conditions. Oscillations are well represented, and the temperature is on an appropriate range in the case of irradiation disturbances. In this model, the new temperature difference between the inlet and outlet is obtained from the irradiation, oil flow and previous temperature difference. The model provides the driving force for the simulator, and the speed of the change depends on the operating conditions. The multimodel approach for combining specialised submodels provides additional properties since also equations and delays can be different in different submodels. In the multimodel approach, the working area defined by a separate working point model. [13]

Special situations

The functional relationship between the output variable and the input variable are partly smooth and partly complicated nonlinear [12]. Smooth dependencies can be described easily by linguistic equations (LE). Complicated local structures are efficiently detected by the Fuzzy-ROSA method (FRM) [16]. Thus the cascaded modelling with the LE and FRM combines the advantages of both methods, which can result in a considerable improvement of the quality of the resulting final model. Feasibility of the combined LE–FRM approach was demonstrated by applying it to a solar power plant [12].

Some special situations activate when the control system intervenes to dangerous conditions or operational faults. For example, a part of the collector field may be defocused. The oil flow can also distribute in a very uneven way. Different special situations could be further studied with distributed parameter models.

Distributed parameter models

The lumped parameter models predict the average outlet temperature very well, and disturbances can be detected with the fuzzy part. However, the aim of the control system is to keep the maximum temperature of the loop outlets within desired range. The controllers tuned with the lumped parameter simulators are working well but the actual testing of local disturbances and malfunctioning can be done only with the real system. The changes in maximum temperatures are always faster than the changes in the average temperature. The distributed parameter model was developed for this purpose.

Development of a distributed parameter model for such a nonlinear system is very challenging. All the necessary parameters are not available in changing operating conditions. However, the previously developed lumped parameter models provide a feasible starting point for this new development. The key fact was that the dynamic linguistic equation model needs only the current situation to be able to predict the new outlet temperature. Introducing the LE modelling fits well to the numerical solutions of distributed parameter models presented in [17].

The solution was to divide the collector field model into modules, and apply the dynamic LE models in a distributed way. Equation (6) is modified by including the oil flow F to the model:

$$\frac{T_i(t+\Delta t) - T_i(t)}{\Delta t} = a_1 I_{eff} + a_2 T_i(t) + a_3 T_{amb} + a_4 F \tag{8}$$

The previously developed lumped parameter models calculate the temperature difference over the whole collector field. Here the temperature change in a volume element is fraction defined by the flow rate, i.e. the result would be the same as with the lumped parameter models if the operating conditions are exactly same throughout the collector field.

In cloudy conditions, the heating effect can be strongly uneven. These effects are simulated by introducing disturbances into the irradiation. The flow

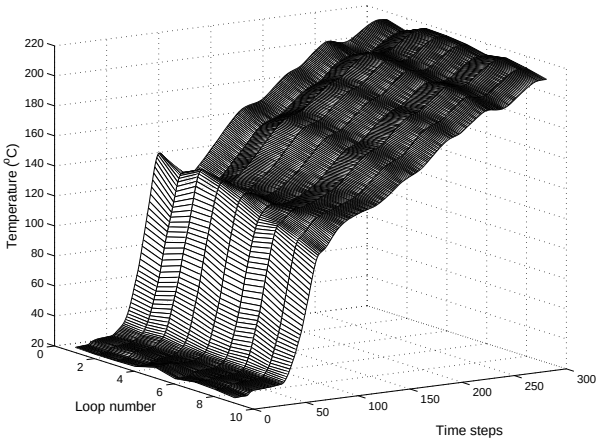


Figure 11: Temperatures of the collector loops in startup phase (June 21, 2002).

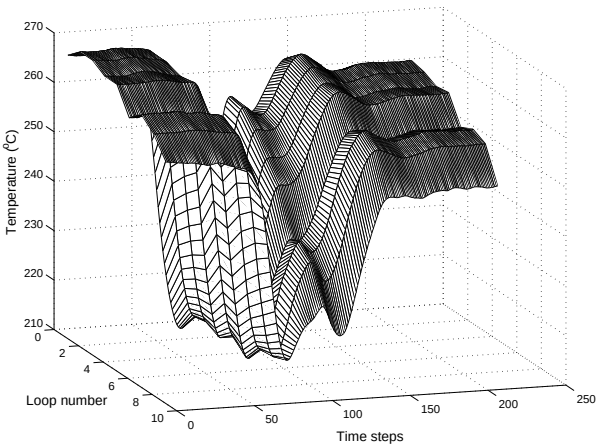


Figure 12: Temperatures of the collector loops during a load disturbance (June 21, 2002).

rate depends also on the density that is decreasing with increasing temperature. Uneven distributions of the oil flow are important if the oil flow changes are rapid since some loops may be unable to follow. The flow of the oil is changed rapidly in the startup phase to stop the fast increase of the outlet temperature. The temperature increases slower in the loops which far from the storage system (Figure 11). As the maximum temperature was controlled, the controller stops the fast the fast temperature increase with high oil flow. The highest temperature drop after the flow peak, and the differences between loop temperatures keep quite small after this even when the flow is again much lower. Obviously, the flow of the oil is unevenly distributed if the flow is very low, and especially if the flow is circulating only in the field. As the properties of the oil change very quickly in these temperature (Figure 3), many disturbances may occur.

A strong load disturbance has similar effects as the startup: the differences between loop temperatures increase (Figure 4), and the temperature is changing faster in the loops which are close to the storage system (Figure 12).

Cloudy conditions seem to reduce the temperature differences between loops both in the startup phase (Figure 13) and in the normal operation (Figure 14). Obviously, big differences cannot evolve as the heating effect is not so high. Another reason is that the spots of high irradiation are continuously moving in the field.

The distributed parameter model was tested by com-

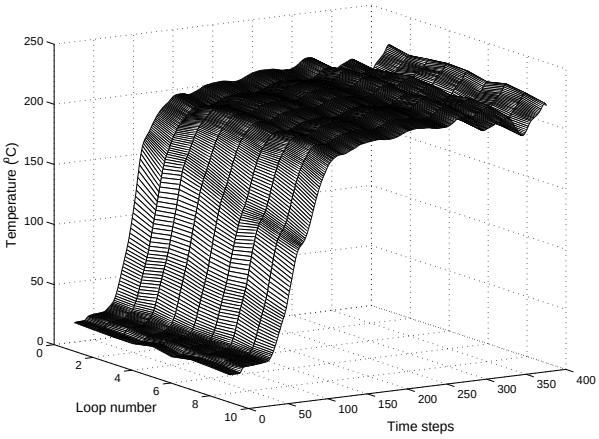


Figure 13: Temperatures of the collector loops in startup phase in cloudy conditions (June 12, 2002).

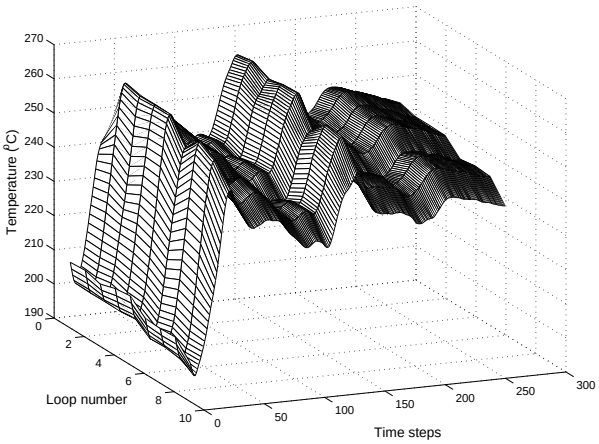


Figure 14: Temperatures of the collector loops in cloudy conditions (June 12, 2002).

paring different scenarios on the distribution of the oil between the loops in smooth irradiation conditions. The cloudy conditions were studied by introducing random fluctuations on the local irradiation in the collector field.

In the control design, the dynamic simulation models are used instead of the real process. The measured variable, the outlet temperature, is generated by the simulator, and the controller changes the flow of the oil to get the outlet temperature to the setpoint. The inlet temperature starts from the ambient temperature, and later the temperature of the storage tank is taken into account. Same simulated or collected measurements, e.g. solar irradiation, are used in the controller and in the simulator.

Discussion

The dynamic LE simulator is a practical tool in the controller design. The resulting controller combines smoothly various control strategies into a compact single controller. The controller takes care of the actual set points of the temperature. The operation is very robust in difficult conditions: startup and set point tracking are fast and accurate in variable radiation conditions; the controller can handle efficiently even multiple disturbances. Adaptive set point procedure and feed forward features are essential for avoiding overheating. The new adaptive technique has reduced considerably temperature differences between collector loops. Efficient energy collection was achieved even in variable operating condition [7].

Exact comparison of the results of the distributed parameter model with the real system cannot be done since there are no measurements on flow distribution in the collector field. There is only one measurement location for the irradiation, and actually, this measurement location is outside the field. This would be a problem for evaluating the performance of the simulation model but the model is clearly suitable for the control design. Different situations can be generated for the controllers to handle.

Disturbances during the normal operation are handled very well but this was done fairly well even with the lumped parameter models. The actual benefit is achieved in modelling of the startup and load disturbances. In these cases, also the density of the oil changes considerably.

The solar collector field is a small test case for a methodological extension where nonlinear intelligent models are distributed into volume elements. These dynamic LE models were similar to those used in dynamic lumped parameter system, and the lumped parameter solution corresponds to the situation where the loops are operating identically, the flow is evenly distributed between the loops, and the irradiation are distributed evenly throughout the collector field.

Conclusions

The dynamic models provide a smooth and accurate overall behaviour achieved with linguistic equations combined special situation handling with fuzzy systems. The new adaptive control technique tuned with this simulator has reduced considerably temperature differences between collector loops. Efficient energy collection was achieved even in variable operating condition. The new distributed parameter model is also aimed for control design. It extends the operability of the simulator to evaluating the controller performance for drastic changes, e.g. startup and large load disturbances, and local disturbances and malfunctioning. This extension is important for controlling the maximum temperature of the collector field as the previous models were capable only for the simulation of the average temperature.

References

[1] Juuso E. K., P. Balsa, and K. Leiviska. Linguistic Equation Controller Applied to a Solar Collectors Field. In *Proceedings of the European Control Conference -ECC'97, Brussels, July 1 - 4, 1997*, volume Volume 5, TH-E I4, paper 267 (CD-ROM), 6 pp., 1997.

[2] Juuso E. K., P. Balsa, and L. Valenzuela. Multilevel linguistic equation controller applied to a 1 mw solar power plant. In *Proceedings of the ACC'98, Philadelphia, PA, June 24–26, 1998*, volume 6, pp. 3891–3895. ACC, 1998.

[3] Meaburn A., *Modelling and Control of a Distributed Solar Collector Field. Ph.D Thesis.* UMIST, 1995.

[4] Zarza E., *Solar Desalination Project Phase II Results and Final Project Report*. CIEMAT, 1995.

[5] Camacho E., M. Berenguel and F. R. Rubio, *Adaptive Control of Solar Plants.* Springer, London, 1997.

[6] Juuso E. K., Fuzzy control in process industry: The linguistic equation approach. In H. B. Verbruggen, H.-J. Zimmermann, and R. Babuska, editors, *Fuzzy Algorithms for Control, International Series in Intelligent Technologies*, pp. 243–300. Kluwer, Boston, 1999.

[7] Juuso E. K. and L. Valenzuela, Adaptive intelligent control of a solar collector field. In *Proceedings of Eunite 2003 - European Symposium on Intelligent Technologies, Hybrid Systems and their implementation on Smart Adaptive Systems, July 10-11, 2003, Oulu, Finland*, pp. 26–35. Wissenschaftsverlag Mainz, Aachen, 2003.

[8] Camacho E., F. Rubio and F. Hughes, Self-tuning control of a solar power plant with a distributed collector field. *IEEE System Control Magazine*, 1992 (April):72–78.

[9] Valenzuela L. and P. Balsa. Series and Parallel Feedforward Control Schemes to Regulate the Operation of a Solar Collector Field. In *Proceedings of the 2nd User Workshop Training and Mobility of Researchers Programme at Plataforma Solar de Almeria, November 10-11, 1998*, pp. 19–24. Ciemat, 1998.

[10] Neves da Silva R., Model based predictive control with time-scaling of a Solar Field. In *Proceedings of the 2nd User Workshop Training and Mobility of Researchers Programme at Plataforma Solar de Almeria, November 10-11, 1998*, pp. 13–18. Ciemat, 1998.

[11] Farkas I. and I. Vajk, Internal model-based controller for a solar plant. In *CD-ROM Preprints of the 15th Triennial World Congress*, IFAC, Barcelona, Spain, July 21-26, 2002, 6 pp.

[12] Juuso E. K., D. Schauten, T. Slawinski, and H. Kiendl, Combination of linguistic equations and the fuzzy-rosa method in dynamic simulation of a solar collector field. In L. Yliniemi and E. Juuso, editors, *Proceedings of TOOLMET 2000 Symposium - Tool Environments and Development Methods for Intelligent Systems, Oulu, April 13-14, 2000*, pp. 63–77, Oulu, 2000. Oulun yliopistopaino.

[13] Juuso E. K., Intelligent dynamic simulation of a solar collector field. In A. Verbraeck and V. Hlupic, editors, *Simulation in Industry, 15th European Simulation Symposium ESS 2003, Delft, The Netherlands, October 26-29, 2003*, pp. 443–449, Erlangen, 2003, Gruner Druck 2003.

[14] Juuso E. K., Integration of Intelligent Systems in Development of Smart Adaptive Systems. *International Journal of Approximate Reasoning*, 35:307–337, 2004.

[15] Juuso E. K., Hybrid Intelligent System in Dynamic Simulation of a Solar Collector Field. In B. Elmegaard, N. Houbak, A. Jakobsen F. J. Wagner, editors, *Proceedings of SIMS 2000 Conference, Lungby, Denmark, September 18-19, 2000*, pp. 267–276, Lungby, 2000. SIMS and DTU.

[16] Kiendl H., Design of advanced fuzzy systems. In *Proceedings of TOOLMET'99 Symposium-Tool Enviroments for Intelligent Systems*, pp. 57–76, Oulu, Finland, 1999.

[17] Juuso E. K., Multilayer simulation of heat flow in a submerged arc furnace used in the production of ferroalloys. In M. Amouroux and A. El Jai, editors, *Control of Distributed Parameter Systems 1989. Selected Papers from the 5th IFAC Symposium, Perpignan, France 26-29 June 1989, IFAC Symposia Series, 1990, Number 3*, pp. 459–464, Oxford, UK, 1990. Pergamon.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

COMPUTER-BASED DESIGN TOOL FOR GEOTHERMAL PIPING SYSTEMS

Anna Karlsdottir*, Fjola Jonsdottir and Magnus Thor Jonsson
Department of Mechanical Engineering
University of Iceland
Reykjavik, Iceland

Abstract

The objective of this study is to develop a design system for geothermal piping systems. Designing geothermal piping systems requires expert skills, built on designers experience. This expert knowledge has to be accessible for the organization, as well as less experienced designers. By developing a design system, it is possible to lead the designer through the design process; expert knowledge will be available at all times and the design procedure will be more automatic, resulting in shorter design time. In this study, a design system called *GeoMind* is developed. The system is tested by designing a new piping system, at Iceland's *Hellisheidi Power Plant*. The results show that the design process is more accessible and automatic, the designer is led through the design process, all which result in a shorter design time.

Keywords: Design system, geothermal piping systems.

Introduction

The demand of utilizing reusable energy sources, such as geothermal power, is increasing around the world. The design of a geothermal power plant is a complex distributed problem, and the design of geothermal piping systems is an important factor in the design.

Today the design is built upon the designers' experience and is mostly ad-hoc design for every geothermal area. The design process can be time consuming and it is difficult, even for the experienced designer, to have an overview of all the different design options. Therefore, a design system for the design of geothermal piping systems can save the designer considerable time and keep track of different solutions. The purpose of this project is to develop a computer-based design system. Currently, such a system, which takes into account different factors involved in the design, is not available on the market for geothermal piping system.

Generally, geothermal power plants are situated in active seismic areas, and therefore, seismic design is a major factor in the structural design of geothermal piping systems. The focus here is on the design of piping systems that lead the geothermal fluid from the wells to the separation station. The geothermal piping system has to be flexible enough to allow thermal expansion but also stiff enough to withstand the seismic load action.

It has been shown that part of the design process can be automated [1]; therefore, it is interesting to study how the designers' experience can be utilized to make the whole design process more automatic, making a computer-based design system.

Many computer-based design systems have been developed, especially on a single computer and now researches are looking into distributed computer-based design system utilizing client-server architecture [2]. Agent-based integrated design systems have been investigated, for example, in aerospace vehicle design, where the level of automation is provided by an expert system [3].

Structure of the design system

Design of piping systems in geothermal areas is a complex, coupled design problem. In order to develop a design system for geothermal piping sys-

*Corresponding author. Phone: +354 585 9203, Fax: +354 585 9299, E-mail: anna@marorka.com

tems, the details of the design process have to be analyzed and the appropriate architecture for the design system stated. The model for the design process is shown in Figure 1.

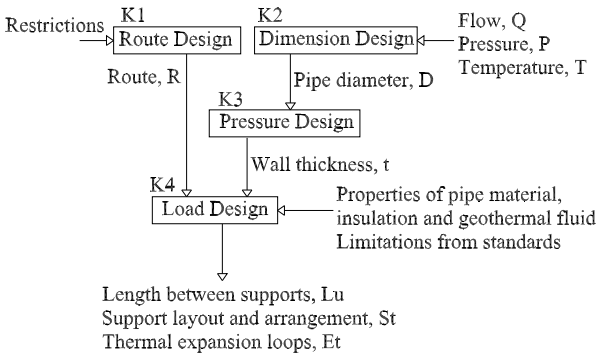


Figure 1: Model for geothermal piping system.

The product developed is called *GeoMind*. The name describes the purpose of the design system; *Geo* refers to geothermal areas and *Mind* refers to the knowledge captured in the system.

GeoMind can communicate with other programs, for example, finite element analysis and CAD software, using application programming interface, *API*. *GeoMind*'s output is an XML-document. The XML is a text based standard for representing data in a structured format, and it can be used on any platform and for many applications in different languages.

There is no limitation on the applications or programs that can communicate with *GeoMind*, only the application programming interface for each program has to be tailor made according to the program in question. The application programming interface reads the XML-document and converts the information into a suitable form for the program that is required. The programming language C# [4] was used to develop *GeoMind*.

The design process for geothermal piping systems can be divided into four main subsystems: route selection, dimension design, pressure design and load design, as shown in Figure 1. The route of the piping system is dependent on the environment, that is, landscape and other structures. The pipes can either be above ground or underground. The dimension design gives the pipe diameter which is calculated for a certain mass flow of geothermal fluid and the pressure drop in the pipe. The pressure design gives the pressure class and hence the necessary wall thickness. Finally, the load design involves the stress analysis where the layout and types of supports are decided, keeping stresses below allowable limits. The support arrangement influences natural frequencies and mode shapes of the piping system, due to the simultaneous earthquake motion and high thermal loading, and is a large factor in load design for geothermal piping systems. Trial and error processes have been utilized, achieving the optimum support positions and arrangement, but studies have shown that a genetic algorithm can been used to arrange supports automatically [1].

Standards that are used in the design system are Ansi B31.1 [5] for pressure piping design, Eurocode 8 [6] for seismic design and IST12 [7] for snow and wind load design.

The subsystems interact with each other as shown in Figure 1. Each subsystem, or module, is designed to minimize the programming interaction with other modules. If it is necessary to modify a module, the modification will not affect other modules. *Load design* requires both input from *Route Selection*, *Dimension Design* and *Pressure Design*.

Route Selection is dependent on the landscape and legal issues. *GeoMind* does not have an automatic process for laying the pipe route, but it offers the user to open a CAD model and select the piping system's route from the CAD model. *GeoMind* then generates a file containing the end points and the section points of the route. This procedure requires application programming interface as described earlier. If the application programming interface for CAD programs is not available, *GeoMind* can read a coordination file directly.

Dimension Design refers to the selection of pipe diameter and depends on the pressure drop and mass flow of the fluid inside the pipe. Since the focus here is on two-phase flow geothermal pipelines, the fluid flow and velocity is used to determine the pipe diameter, and the effect of pressure drop is ignored. Dimension Design returns minimum pipe diameter required for transporting the geothermal fluid at given conditions.

Pressure Design refers to the calculation of minimum wall-thickness required. The pipe wall has to be thick enough to withstand the pressure inside the pipe. The required wall thickness is determined according to Ansi B31.1 [5].

Load Design refers to stresses developed in the pip-

ing system due to internal and external loading. The Load Design gives the layout for thermal expansion loops, location of supports and the types of supports. The stresses are controlled with different types of supports, that constrain the motion of the pipe. The following are the most common types of supports used in geothermal piping systems. Anchors are fixed supports which neither allow displacements nor rotation. They are used to divide large piping systems into smaller sections, and are therefore always at the end points of the piping system sections. Free supports hinder motion in vertical direction. In order to control the transverse motion of the piping system, longitudinally guided supports are used. They allow axial displacement, but no transverse or vertical displacement. Sometimes, it is necessary to restrict the longitudinal displacement of the pipe, but allow transverse displacements. In these cases, transversely guided supports are used. Finally there are rotationally guided supports, which allow rotation but not displacements.

The pipe has to be allowed to expand because of the thermal expansion, but has to be stiff in order to withstand the vertically and horizontally applied loads. The vertical loads that act on the pipeline are the self-weight of the pipe, the fluid weight, the snow load and the vertical component of the seismic load. The horizontal loads are the wind load and the horizontal component of the seismic load.

The distance required between supports is calculated due to loads acting on the pipeline. Supports may not be located under bends, because there the piping system has to be able to expand. Pipe bends are more flexible than straight pipes and therefore the support has to be positioned as near the bend as possible in order to support the pipe without restricting the expansion.

The start guess for the support arrangement for the pipeline is as follows. The location of deflection points of the pipeline is known, and the supports between them are located a distance L apart, where L is the required length between supports. The types of the supports are set in the following manner: end points are fixed supports, where neither displacements nor rotation is allowed, other supports hinder vertical motion and are guided longitudinally. Supports that support bends, only hinder vertical motion, allowing thermal expansion.

Stresses in the pipeline may not exceed allowable stresses given by Ansi B31.1 [5]. In order to check the stresses in the pipeline, a stress analysis has to be carried out. *GeoMind* is able to communicate with stress analysis software as described earlier.

In this study, a module for automatic support arrangement was developed. If stress analysis indicates that the stresses in the pipeline are above the allowable stress limits, the automatic support arrangement changes the types of supports, in order to reduce the stresses in the pipeline below allowable stress limits. Then, the new support arrangement is then analyzed again. The usage of the automatic module demonstrates that an optimization module can be attached to the design system.

GeoMind's output is an XML-document and the communication with a stress analysis application requires application programming interface, tailor made for the finite element analysis application in question. The application programming interface takes the information in the XML-document and converts it into readable form for the finite element analysis application.

Architecture and usage

Architecture of the design system, *GeoMind*, is shown in Figure 2.

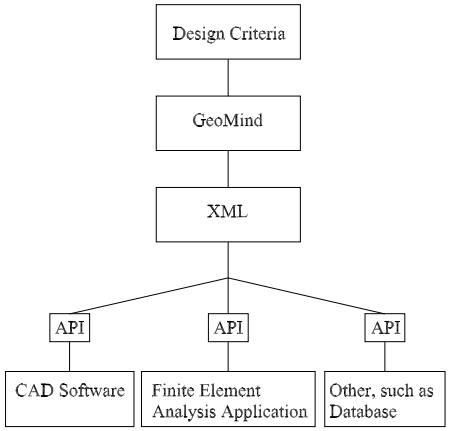


Figure 2: Architecture of the design system *GeoMind*.

The design criteria involves properties of the pipe material, the geothermal fluid and the insulation, constants regarding the geothermal area and the pipe route. The design system takes the input information and designs the piping system. In order to complete

the design, *GeoMind* has to communicate with a finite element analysis software for the stress analysis part. *GeoMind*'s output is an XML-document, where all the information about the piping system is available for the application programming interface, which will convert the data into a mode acceptable for the finite element analysis application. The XML-document includes every detail of the piping system and can be interpreted by the appropriate application programming interface for the program connected to *GeoMind*.

GeoMind enables engineers to design geothermal piping systems in the same computer application. *GeoMind* contains all the information needed for the design, such as, standards and design documents. *GeoMind* includes knowledge formed by designers and *GeoMind* can communicate with other software needed for the design process, such as, CAD systems and finite element analysis programs.

When the designer starts *GeoMind*, the front site appears; see Figure 3.

Figure 3: Front site of *GeoMind*.

There it is possible to choose between different actions, ROUTE DESIGN, PIPE DESIGN and INFORMATION.

INFORMATION will lead the designer to a site that contains design documents and standards available for pipe design. The newest design documents and standards should always be updated in the design system.

ROUTE DESIGN leads the designer to a site where he can locate a CAD drawing for the pipe route. There, the designer can generate a file containing the coordinates of the pipe from the drawing. When the coordinates' file is available, the designer can go to PIPE DESIGN to design the piping system. This process requires an application programming interface for the CAD software used.

PIPE DESIGN site is shown in Figure 4. Here, the designer has to set the design criteria required and will get the minimum pipe diameter and the minimum wall thickness required for the project by clicking *Dimension and Pressure Design*. The designer can select manufactured pipe from pipe catalogs and enter the standard pipe diameter and wall thickness into the *Selected Pipe Diameter* and *Selected Wall Thickness* text boxes, and use those values in following design process.

The final design of the piping system is generated by selecting *Stress Analysis*. The *Stress Analysis* can not be selected if the *Dimension and Pressure Design* has not been completed. The piping system's route is needed for *Stress Analysis*. If the application programming interface for CAD software is not available, the designer enters the location of a coordination file containing the coordinations of piping system's endpoints and section points into *Pipe Route Coordinates File*.

The final design of the piping system is shown in Figure 5. The output is various depending on the application programming interfaces used. If the iteration process in the stress analysis is automatic, stress analysis figures exist, if the iteration process is made manually by the designer, the model for the finite element analysis application is available. The layout of the supports and types can be displayed in the CAD drawing, and the design information can be stored in a database. The XML-document, containing information of the final design for the piping system, is always accessible for the designer from FINAL DESIGN.

Case study

To test the relationship between *GeoMind* and an external program, the professional finite element analysis software ANSYS [8] was used for stress analysis.

The application programming interface developed converts the XML-document generated by *GeoMind* into a model for ANSYS. ANSYS is then automatically started where the model is analyzed and the results sent back to the application programming in-

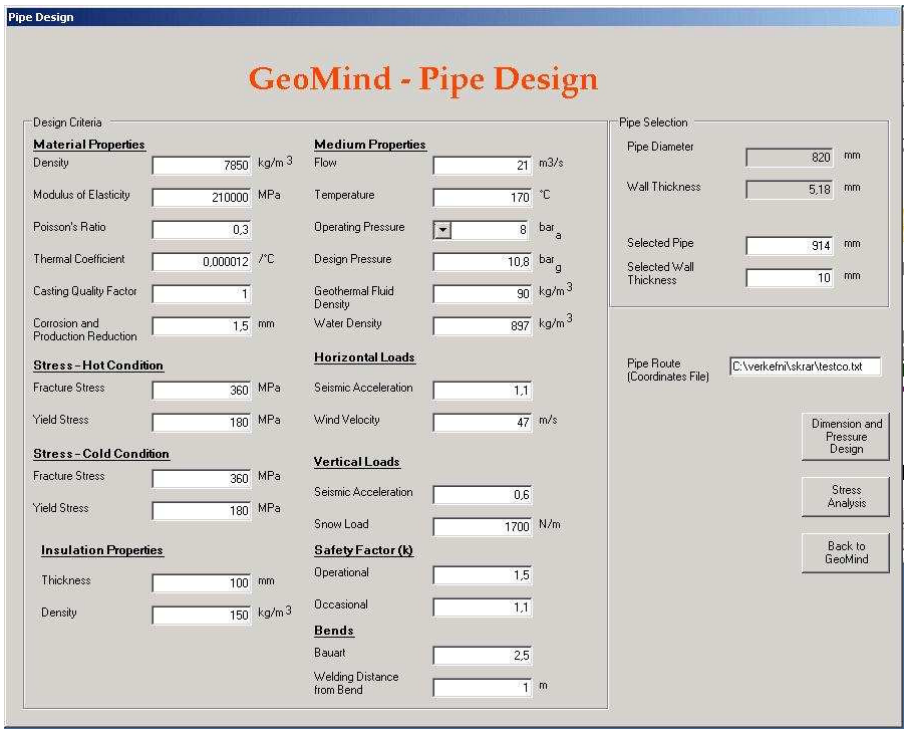


Figure 4: PIPE DESIGN site in *GeoMind*.

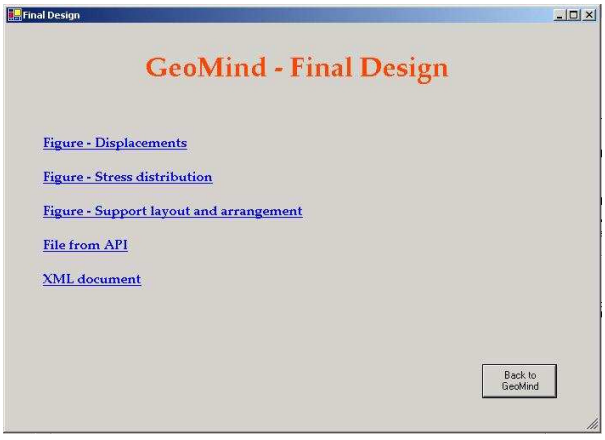


Figure 5: FINAL DESIGN site in *GeoMind*.

terface which translates the output file into XML-document read by *GeoMind*. *GeoMind* translates the XML-document, reads maximum stress values and relative supports' numbers, and then runs the automatic support arrangement module, which generates a new support arrangement, if the stresses are above the allowable stress limits. Then *GeoMind* produces a new XML-document which is sent to the *GeoMind* - ANSYS application programming interface and the procedure is repeated. The iteration continues until the stresses in the piping system are below the allowable stress value or until the iterations have reached a limit set by the user. The architecture of *GeoMind* gives the opportunity to replace the automatic support arrangement module for an optimization module in the future. The procedure described above is shown in Figure 6.

The design task is to design a new piping system for the *Hellisheidi Power Plant*. The route is not pinned down, only a general route is known. The general route of the piping system is shown in Figure 7.

The piping system is made of straight pipes connected with 90 degrees bends. There are two thermal loops in the piping system and the end points are fixed.

The design process for *Dimension and Pressure De-*

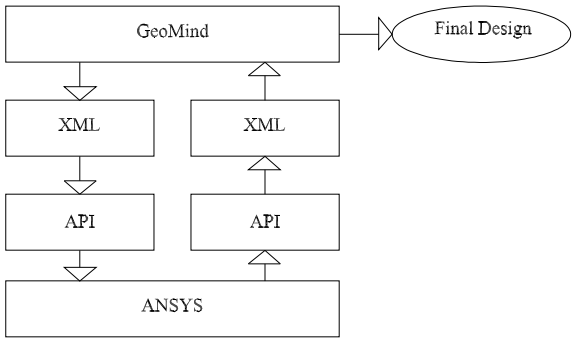


Figure 6: Design procedure using application programming interface with automatic connections with ANSYS.

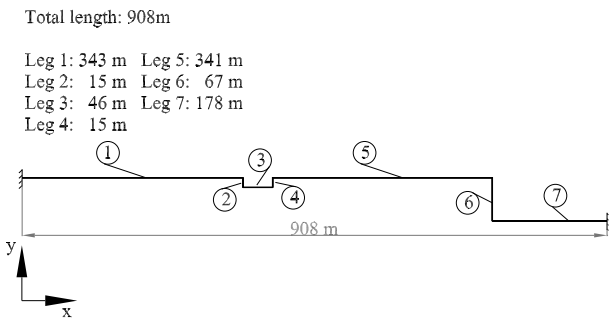


Figure 7: Route proposal at *Hellisheiði Power Plant*.

sign is described earlier. The calculations in *Dimension and Pressure Design* give minimum diameter 840 *mm* and minimum wall thickness 5.2 *mm*. In the following design, the manufactured standard pipe chosen is *DN*900, which has a pipe diameter of 914 *mm* and the wall thickness is 10 *mm*.

The automatic support arrangement does not give stresses in the pipeline below allowable stress limits. The result indicates that the pipe layout has to be changed, to be able to reduce the stresses in the piping system. One way to reduce the stresses below allowable stress limits, is to divide the piping system into two systems using a fixed support.

New coordinates files are made, dividing the piping system into two systems, one thermal expansion loop in each. *GeoMind* is used to design the two piping systems. The stress analysis for both piping systems give stresses in the pipelines below allowable stress limits. The support arrangement for the two piping systems is shown in Figure 8.

Conclusions and further work

A design tool for piping systems in geothermal areas has been developed in this study.

In an organization where expert knowledge is the main asset, it is vital to be able to manage the expert knowledge and making it available and easily accessible for the employees. That can be done by using a design system, such as *GeoMind*.

In order to develop the design system, the design process was analyzed and divided into subsystems or modules, which are loosely coupled. By building the design system up in modules, opens the possibility to add new modules to the system. Of special interest is an optimization module regarding thermal expansion loops and support arrangement. From the design system analysis, an architecture for a knowledge based design system was presented and *GeoMind*, a design system, programmed. *GeoMind* has three main modules, INFORMATION, ROUTE DESIGN and PIPE DESIGN.

INFORMATION contains all the documents related to the design process; that is, design documents, CAD models, standard documents, pipe manufacture catalogs to mention a few. In INFORMATION the user can find every document related to the design. The responsibility of update of the documents in INFORMATION has to be allocated to an employee in the

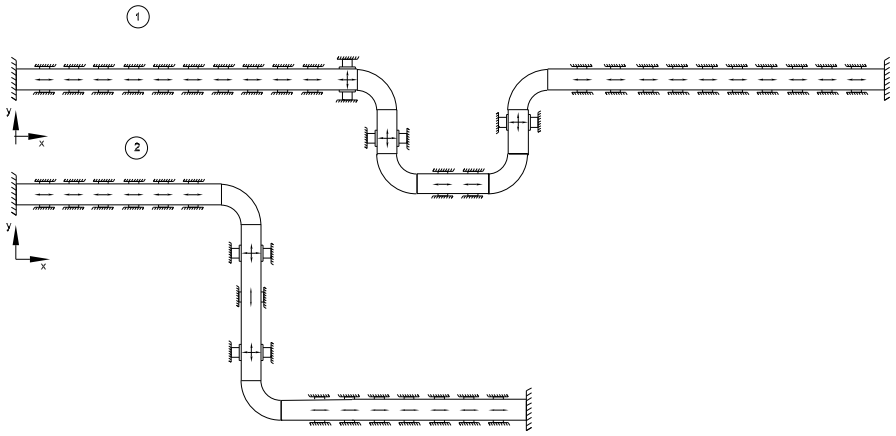


Figure 8: Support layout and arrangement for two the piping systems at *Hellisheidi Power Plant*.

organization using *GeoMind*.

ROUTE DESIGN is for designing the pipe route for the piping system. It can be connected to a CAD software, using appropriate application programming interface. Then, it is possible to generate a file, containing the coordinates of the end points and section points, describing the piping system. If this feature is not used, *GeoMind* reads a file containing the end points and section points coordinates of the piping system.

PIPE DESIGN is the module where the pipe is designed. The input is design criteria for the geothermal fluid, pipe material and the insulation material, as well as limitations from standards. PIPE DESIGN's output is twofold; *Dimension and Pressure Design* gives required pipe diameter and wall thickness, and *Stress Analysis* gives the final design of the piping system. Finite element analysis software has to be used for analyze the stresses in the piping system. An application programming interface has to be made for the interaction between *GeoMind* and the finite element analysis application, where the application programming interface converts the XML-document generated by *GeoMind* into an understandable form for the finite element analysis application. It is up to the user how the application programming interface is made.

Here, an application programming interface was developed using the stress analysis application ANSYS [8]. The application programming interface started ANSYS automatically, where the design information was analyzed, the stress analysis data, generated by ANSYS, was then converted into XML-document read by *GeoMind*, where the support arrangement was made again, if necessary, according to the stress analysis data, using the automatic support arrangement module.

A design system for designing geothermal piping systems, makes the design problem more accessible. All documents related to the design is accessible at one site. The designer is led through the design, first asked to design the piping system route, then to collect the design criteria for the piping system and the geothermal area. The design system calculates minimum pipe diameter and wall thickness required according the to design criteria. The designer can select manufactured pipe, from pipe catalogs, and the standard pipe diameter and wall thickness will be used in the remaining design process. Stress analysis has to be made for the final design of the piping system. Final design information is easily accessible and can be published in various ways. It is also possible to store the final design information in a database, if required, making the design information accessible for every designer in an organization.

The use of *GeoMind* will make the design less troublesome, because all the design documents and criteria are accessible in one location. The design is

more automatic and the possibility of connecting different modules, such as different finite element analysis applications or optimization algorithms, to the design system makes the design more flexible. *GeoMind* will reduce the design time, resulting in reduced design cost.

In the future, the design system should be designed as a distributed system, a server-client system, where the several users can work with a designer system interface and the applications needed for the design, i.e. finite element analysis software, CAD programs and optimization packages, are stationed at servers. Case studies in distributed design have shown that computational time is reduced significantly; heterogenous software and platform are integrated and designers work collaboratively [9]. Agent-based system could be applicable, where the agents would control the access to the applications on different servers.

It is of interest to design and optimization module for the equilibrium between the thermal expansion and load acting on the geothermal piping system.

The ROUTE DESIGN module could offer an artificial intelligence model, where the machine would recognize restrictions in the landscape and know other restrictions, such as legal issues and environmental protection.

Acknowledgment

The support for this study was provided by the University of Iceland Research Fund and Reykjavik Energy. This support is gratefully acknowledged. The study was done in collaboration with *VGK Consulting Engineers*, Iceland.

References

[1] Ágústsson, G., 2002, *Seismic Design of Geothermal Piping Systems*, Master Thesis, University of Iceland, Faculty of Engineering.

[2] Xue, D., Xu, Y., 2001, *Web-Based Distributed System and Database Modeling for Concurrent Design*, Proc. of the ASME Design Engineering Technical Conferences, Pittsburgh, PA.

[3] Gee, K., 2001, *Vehicle Analysis Using an Agent-Based Analysis Tool*, Proc. of the ASME Design Engineering Technical Conferences, Pittsburgh, PA.

[4] Microsoft Corporation, 1987-2001, *Microsoft .net Framework 1.0, Visual C#.net,* Version 1.0.3705.

[5] ANSI B31.1-2001, *ASME Code Pressure Piping, American National Standard.*

[6] Eurocode 8, FS ENV:1998-4:1998, *Design of Structures for Earthquake Resistance of Structures- Part4: Silos, Tanks and Pipelines.*

[7] ÍST 12, 1989, *Loads for the Design of Structures*, 3rd edition, (in Icelandic), Icelandic Standards

[8] Ansys, Theory Manual, v.5.7. Ansysi Inc., Canonsburg, PA 15317.

[9] Xiao, A., Choi, H.J., Kulkarni, R., Allen, J., Rosen, D., Mistree, F., 2001, *A Web-based Distributed Product Realization Environment*,Proc. of the ASME Design Engineering Technical Conferences, Pittsburgh, PA

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

PROPOSAL OF A SIMPLE POLYMER ELECTROLYTE MEMBRANE FUEL CELL (PEMFC) STAGES ASSESSMENT METHOD

Fabio Lo Mastro
Dipartimento di Meccanica
Università degli Studi di Palermo
lomastro@dima.unipa.it

Gandolfo Blando
Dipartimento di Meccanica
Università degli Studi di Palermo
gandalef@tin.it

Abstract

The purpose of this paper is to explain a model that can predict the Polymer Electrolyte membrane Fuel Cell (PEMFC) system fuel consumption, and more heat and electricity production according to the load. Vice versa it is possible to predict with such model the power produced knowing the PEMFC system fuel consumption. Structure and validation of the model, including theory, and preliminary results demonstrating the model's use with the building of a simulation program, will be presented. Many mathematic models can be found in the literature that evaluate electrochemical fuel cell performance and that seem to produce excellent results, however, the aim here is to create a simple evaluation model for fuel cell which can be integrated into commercial simulation modular codes.

Keyword

PEM Fuel Cell, Simulation, Modelling

Nomenclature

$\dot{m}_a$	Mass flow rate of water into reformer [kg/s]
$\dot{m}_{H_2}$	Mass flow rate of hydrogen [kg/s]
α	exchange coefficient
ΔG	Gibbs free energy change [kJ]
ΔG_T	Gibbs free energy change at temperature T [kJ]
ΔH_{FP}	Enthalpy change of products formation [kJ]
ΔH_{FR}	Enthalpy change of reagents formation [kJ]
ΔH_{FW}	Enthalpy change of H_2O (vap) formation [kJ]
ΔH_R	Enthalpy change of reforming reaction [kJ]
ε_c	AC/DC converter efficiency [%]
ε_{el}	Electrical generation efficiency [%]
ε_f	Fuel utilisation efficiency [%]
ε_h	Heat efficiency [%]
ε_{hex}	Heat exchanger effectiveness [%]
ε_{hp}	Combined heat and power efficiency [%]

ε_i	Current efficiency [%]
$\varepsilon_{\max}$	Maximun theoretical efficiency [%]
ε_R	Reformer efficiency [%]
ε_v	Voltage efficiency [%]
A_c	Cell area [m^2]
b	Tafel slope for oxygen reduction [V]
i	Current density [A/m^2]
F	Faraday constant [C/kmol]
H_{H_2}	Heating value of H_2 [kJ/kg]
I_{st}	Operative stack current [A]
I_T	Cell currents [A]
K^*	Equilibrium constant of reaction at standard temperature and pressure
K_p	Equilibrium constant of reaction at temperature T
L	Losses [W]
m	mass transport coefficient [V]
n_e	Number of electrons
N_T	Number of cells
p_{H_2}	Hydrogen partial pressure [bar]
p_{O_2}	Oxygen partial pressure [bar]
p_{H_2O}	Water partial pressure [bar]
Q_{in}	Heat input [W]
Q_{out}	Heat output [W]
q_m	Charge per unit mass
v_{H_2}	Hydrogen stoichiometric coefficient
v_{O_2}	Oxygen stoichiometric coefficient
v_{H_2O}	Water stoichiometric coefficient
R	Perfect gas constant [kJ/kmol K]
R_i	Ohmic resistance of the cell [Ωm^2]
S	Active cell area [m^2]
T	Temperature [K]
V	Cell voltage [V]
V_0	Reversible cell voltage [V]
V_{st}	Operative stack voltage [V]
W_{max}	Maximum electrical power [W]
W_R	Reformer power [W]
W_S	Stack power [W]
W_{DC}	DC power [W]
W_{AC}	AC power [W]
z	Valency

1. Introduction

A fuel cell is a device that converts chemical energy directly into electrical one and heat without combustion. This conversion is facilitated by an electrode-electrolyte structure that operates on principles similar to chemical batteries. A typical scheme of a fuel cell can be seen in Figure 1. The electrolyte is usually what the particular fuel cell is named after, such as in the case of PEM the electrolyte is a thin ion conducting membrane. However, while a battery's fuel and oxidant supplies are stored within the cell, fuel cells permit fuel and oxidants to continuously flow through the cell. This is advantageous, because a fuel cell need not be taken offline to be refuelled. PEMFC systems consist also of several subsystems, these include: the fuel cell stack, the fuel processor, and auxiliary systems required for operation.

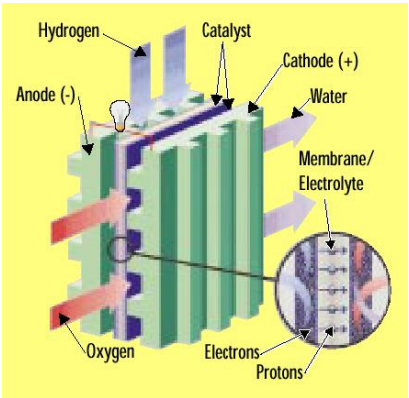


Figure 1: Fuel cell scheme

The purpose of this paper is to explain a model that can predict the PEMFC system fuel consumption, and more heat and electricity production according to the load. Vice versa it is possible with such model to predict the power produced knowing the fuel cell system fuel consumption.
While many fuels may react with oxygen to produce electricity directly, hydrogen has the highest electrochemical potential and yields the highest theoretical fuel conversion. Thus, hydrogen is typically used to fuel low temperature PEM fuel cells. Since natural sources of hydrogen are not available, it must be obtained from a hydrocarbon fuel such as natural gas. In this simulation the fuel used is methane but, as before mentioned, can be

used also other fuels such as methanol, ethanol, pure hydrogen and so on. When the fuel is chosen the program automatically calculates the results.
The model begins by sorting out the reforming and water-gas shift reaction.
After Gibbs energy and enthalpy of FC reaction, it is possible to calculate the electricity and heat output vs. hydrogen consumption. As mentioned above, two options are available on the program:
- calculating the heat output and fuel consumed for a given electrical power output;
- calculating the electricity and heat output for a given supply of fuel.

Since a single fuel cell produces a limited voltage, usually less than 1 volt, in order to produce a useful voltage, a number of fuel cells are connected in series. Series connected fuel cells form a fuel cell stack. In view of the fact that the number of unit cells in a stack depends on the desired voltage, a fuel stack configuration had to be resolved. Two calculation options are available: specify number of cells in series to determine FC stack voltage and number of cells required in parallel to supply power; specify desired FC stack operating voltage to determine numbers of cell required in series and in parallel.
Finally, power conditioning converts the electric power from DC into regulated DC or AC for consumer use.

2. Reforming process

A fuel cell can be fed with pure hydrogen or hydrogen produced on board by a fuel processor. This system converts the primary fuel stored in a tank to a hydrogen rich gas which is fed to the cell. The gas produced by the fuel processor contain carbon monoxide; CO molecules contained in the hydrogen stream cause catalyst poisoning and a significant decrease of the fuel cell performance. In order to decrease the CO content to less than 10 ppm, the fuel processor is composed of three sections; the first is an autothermal process involving a partial oxidation which produces a hydrogen rich stream with CO contents ranging from 6 to 18%. In a second step, a further conversion (shift) of residual CO to CO2 reduces the CO concentration to 0.8-1% with a consequent decrease of CO content. In the third section, a selective oxidation process (prox) reduces the CO concentration to a few ppm.
The reforming is a fuel process that allows to supply relatively pure hydrogen to a fuel cell, using a fuel that is readily available or easily transportable. For houses and stationary power generation, fuels like natural gas or propane are preferred. Methanol, ethanol, and natural gas can be converted to hydrogen in a steam reformer.
In this simulation the fuel used is methane. The model begins by sorting out the reforming and water gas shift reactions. The methane-steam reforming can be modelled as occurring in two stages by two different pathways, the first one (reforming reaction (1)) involves the methane reacts with water steam to form carbon monoxide and hydrogen gases:

$$CH_4 + H_2O \rightarrow CO + 3H_2 \quad (1)$$

followed by a second stage where the water steam splits into hydrogen gas and oxygen (water gas shift reaction (2)), the oxygen combining with the CO to form CO_2 since it's important to eliminate the carbon monoxide from the exhaust stream, because, if the CO passes through the fuel cell, the performance and life of the fuel cell are reduced:

$$CO + H_2O \rightarrow H_2 + CO_2 \quad (2)$$

this gives the overall methane reforming reaction:

$$CH_4 + 2H_2O \rightarrow CO_2 + 4H_2 \quad (3)$$

The overall process is endothermic and therefore requires that external heat is supplied. Excess steam and heat is required to shift the water-gas reaction equilibrium to the right and maximise the hydrogen production from methane.
The model doesn't take into account that neither of these reactions is perfect; some natural gas and carbon monoxide make the FC through without reacting. These are burned in the presence of a catalyst, with a little air; such a catalyst converts most of the remaining CO to CO_2.
The heat required by reforming process is calculated from the enthalpy of reaction [2]:

$$\Delta H_R = \sum \left(\Delta H_{FP} - \Delta H_{FR} \right). \quad (4)$$

3. Gibbs energy and enthalpy of fuel cell reaction

At this point the next step is to sort out the FC reaction and half reactions at the electrodes. For a PEM fuel cell it is:

$$H_2 + \frac{1}{2} O_2 \rightarrow H_2O \qquad \text{overall equation} \quad (5)$$

$$2H^+ + \frac{1}{2} O_2 + 2e^- \rightarrow H_2O \qquad \text{cathode} \quad (6)$$

$$H_2 \rightarrow 2H^+ + 2e^- \qquad \text{anode} \quad (7)$$

These reactions provide also the stoichiometric coefficients and the valency.
By establishing the fuel cell operating temperature *T*, the partial pressures *p* of products and reagents and by taking into account the stoichiometric coefficients, the theoretical electrical molar work of the fuel cell is calculated from the expression:

$$W_{\max} = -\Delta G = -\Delta G_T + RT \cdot \ln \left[\frac{\left(p_{H_2O} \right)^{v_{H_2O}} \cdot \left(p^* \right)^{0,5}}{\left(p_{H_2} \right)^{v_{H_2}} \cdot \left(p_{O_2} \right)^{v_{O_2}}} \right] \quad (8)$$

where [2]:

$$\Delta G_T = RT \cdot \ln \left(K_p \right). \quad (9)$$

At this point the open voltage can be calculated by [2]:

$$V = V_0 - \frac{RT}{zF} \ln \frac{p_{H_2O}^{v_{H_2O}}}{p_{H_2}^{v_{H_2}} \cdot p_{O_2}^{v_{O_2}}} \Rightarrow$$

$$V = \frac{RT}{zF} \left(\ln K^* - \ln \frac{p_{H_2O}}{p_{H_2} \cdot p_{O_2}^{v_{O_2}}} \right). \quad (10)$$

The heat of fuel cell reaction is now calculated by [2]:

$$\Delta H_{FC} = \sum \left(\Delta H_{FP} - \Delta H_{FR} \right). \quad (11)$$

4. Fuel Cell characteristics and efficiency factors

One of the major goals in fuel cell modelling is the prediction of the cells voltage-current characteristics also called the polarisation curves. The voltage-current characteristic is a plot of the measured cell voltage as function of the cells average current density (defined as the measured current output of the cell relative to the active cell area), and therefore it shows the potential power output at different load conditions and is a good indicator of the performance of a fuel cell stack: a high cell voltage corresponds to a high efficiency.
However one should remember that the polarisation curve is a macroscopic quantity, which describes the output characteristics of the cell and does not give detailed information about the microscopic cell performance e.g. local current flux. Having stated this, it is also important to mention that it is an explicit measure of the performance one can expect to obtain. The voltage-current characteristic of a fuel cells system derives principally from two factors (Fig. 2):

- Voltage losses due to internal resistance and electrode activation potentials represented by the *voltage efficiency* [4];
- Current losses due to the effects of fluid flows and concentrations due inturn to the utilisation ratio of the fuel represented by the *current efficiency* [4].

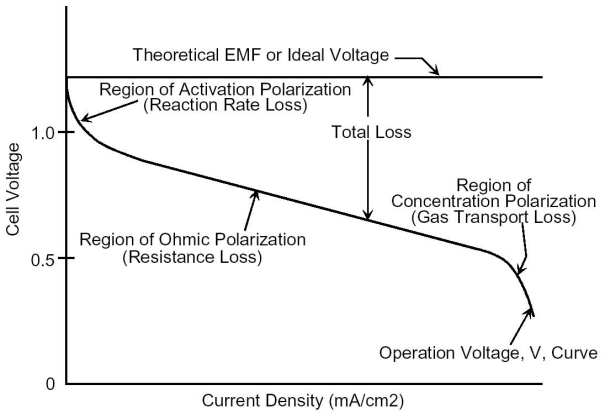


Figure 2: Ideal and Actual Fuel Cell Voltage/Current Characteristic

Internal resistance losses dominate except at high current densities or high fuel utilisation ratios, when mass transfer effects or low concentrations of fuel cause fall off of voltage. Other concentration effects include humidification of the electrodes stream and use of air in place of oxygen. These effects are dependent upon the detailed construction of the fuel cell and the fluid flow channels; because they are hence difficult to model theoretically, the electrochemical energy converter is assumed with two planar, smooth electrodes and the characteristic scale of the feed gas density variation along the channel is vastly greater than the cell thickness. This means that along the channel fluxes of gases and current in porous media can be neglected as compared to the fluxes and current in the plane, perpendicular to the channels so that the complexities of the current versus potential with porous electrodes is avoided.
However, due to the dominance of internal resistance the characteristic is substantially linear over normal operating conditions and may be derived empirically; but a more accurately fuel cell voltage-current characteristic may be represented by an empirical relationship such as that introduced by Kim *et al* [1]:

$$V = V_0 - b\ln i - R_i i - me^{ni} \tag{12}$$

where b is the Tafel slope for oxygen reduction, m and n are parameters that account for the "mass transport overpotential" as a function of current density.
Parameter m affects both the slope of the linear region at the V *vs.* i plot and the current density at which there is a departure from linearity. The value of n instead has major effect in the mass transport limitation region.
Let us calculate now the electrical generation efficiency from the following [2]:

$$\varepsilon_{el} = \left[\frac{W_s}{H_{H_2}\cdot \dot{m}_{H_2}}\right]\cdot\frac{1}{\varepsilon_c} \tag{13}$$

Likewise the other efficiencies can be calculated:

$$\varepsilon_v = \frac{V}{V_0} \tag{14}$$

$$\varepsilon_i = \frac{i\cdot A_c}{\dot{m}_{H_2}\cdot n_e\cdot F} \tag{15}$$

$$\varepsilon_c = \frac{W_{DC}}{W_{AC}} \tag{16}$$

$$\varepsilon_{\max} = \frac{\Delta G}{\Delta H}\,. \tag{17}$$

The previous expression represents the theoretical maximum efficiency of conversion from chemical to electrical energy. In fact, the theoretical maximum amount of electrical energy that can be obtained from the electrochemical reactions occurring in the fuel cell is equal to the change in the Gibbs free energy within the cell, while the total amount of energy released in the electrochemical reactions is equal to the enthalpy change within the cell.
Thus, the theoretical maximum efficiency of conversion from chemical to electrical energy, $\varepsilon_{\max}$, is obtained as the ratio between the change in the Gibbs free energy and the change in the enthalpy that occur in the cell[1].
At this point, it is assumed that a constant DC voltage is required for conversion to a constant AC voltage, the excess DC voltage is dropped through a resistor. Thus, the DC conversion voltage is equal to the fuel cell voltage at rated or maximum required operating current and the voltage efficiency remains constant at partial loads.
Due to the rapid drop off voltage at low concentrations, regulation of fuel flow by voltage could be unstable and regulation of fuel flow proportional to current is assumed. Thus the fuel utilisation ratio or current efficiency is also constant. Ancillary loads such as pumps, fans, pre-

[1] The Second Law of Thermodynamics requires that the theoretical maximum efficiency be achieved only when the cell operates under reversible conditions, and these conditions are approached when there is no electrical load on the stack.

heaters, compressor and so on, are identified and considered.

5. Electricity and heat output vs. fuel consumption

Two options are available:
- calculating the heat output and fuel consumed for a given electrical power output;
- calculating the electricity and heat output for a given supply of fuel.

5.1 Fuel consumed for a required electrical output

- Add ancillary loads referred to AC supply and add converter efficiency factor to give DC output required from fuel cell stack.
- Determine sum of currents through all cells in stack by:

$$\sum I_T = \frac{W \cdot N_T}{V} = \frac{W_S}{V}. \tag{18}$$

- Hence hydrogen utilised in stack from the expression:

$$\dot{m}_{H_2} = \frac{\Sigma I_T}{q_m} \tag{19}$$

- Hence hydrogen flow from utilisation efficiency.
- Fuel flow to reformer from fuel to hydrogen mass ratio and reformer efficiency.
- Total fuel consumption including external pre-heating.

5.2 Electricity output for an available fuel supply

The output of a FC is obtained as follows:
- Total fuel available minus fuel used in external pre-heating.
- Hydrogen flow from reformer by reformer efficiency and hydrogen to fuel mass ratio.
- Hence hydrogen utilised in fuel cell stack from utilisation efficiency.
- Determine sum of currents through all cells in stack from the following:

$$\sum I_T = \dot{m}_{H_2} \cdot q_m \tag{20}$$

$$W = V \cdot \frac{\sum I_T}{N_T} \tag{21}$$

$$W_S = V \cdot \sum I_T \,. \tag{22}$$

- Hence DC output and AC output from converter efficiency.
- Total AC power out less ancillary loads referred to AC output.

6. Heat output

From SFEE [2]:

$$\begin{aligned} &Q_{in} + W_{in} = Q_{out} + W_{out} + \Delta H \Rightarrow \\ &Q_{out} - Q_{in} = \Delta G - \Delta H_R - \Delta H_{FC}\,^2 \end{aligned} \tag{23}$$

Q_{in} includes reformer pre-heating external to reformer or fuel cell stack.
Assuming a fraction x_l of voltage losses within the fuel cell stack is recovered as heat and correcting them trough a fuel utilisation coefficient ε_f, with this assessment the relation (23) can be rewritten:

$$\begin{aligned} &Q_{out} - Q_{in} = (\Delta G - \Delta H_{FC})\varepsilon_f + \\ &+ x_l(\varepsilon_v - 1)\varepsilon_f \Delta G - \Delta H_R \end{aligned} \tag{24}$$

When unutilised fuel is recirculated and burnt in order to preheat the fuel internally:

$$\begin{aligned} &Q_{out} - Q_{in} = (\Delta G - \Delta H_{FC})\varepsilon_f + x_l(\varepsilon_v - 1)\varepsilon_f \Delta G - \\ &- x_l(\varepsilon_v - 1)\varepsilon_f \Delta G - \Delta H_R + (1 - \varepsilon_f)\Delta H_{FW} \end{aligned} \tag{25}$$

The total heat output and the heat gain are reduced by heat losses from the external pre-heater and from losses up to the heat exchanger. The usable

2 ΔG and ΔH_{FC} are negative.

heat output and heat gain are then further reduced by the effectiveness of the heat exchanger. The heat efficiency is calculated as:

$$\varepsilon_h = \frac{(Q_{out} - Q_{in} - L)\varepsilon_{hex}}{\frac{\Delta H_{FC}}{\varepsilon_R}} \quad (26)$$

the denominator is consistent with the electrical generation efficiency.
The heat to power ratio is calculated from:

$$\varepsilon_{hp} = \frac{Q_{out} - L}{W_{\max}}. \quad (27)$$

7. Fuel stack configuration

Two calculation options are available:

- specify number of cells in series to determine fuel cell stack voltage and number of cells required in parallel to supply power;
- specify desired fuel cell stack operating voltage to determine numbers of cells required in series and in parallel.

When the desired operating voltage gives a non integer value of cells in series, an integer value is chosen and the stack voltage is modified. The cell voltage is kept to that originally specified in this case, since a small difference in cell operating voltage could case a large and undesirable change in cell operating current.
When instead the number of cells in parallel to meet the required power is non-integer, an integer value is chosen and the cell operating current adjusted to fit. The resulting operating point may then be compared with the voltage-current characteristic.
Input of the cell thickness, the cell area and the total number of cells enables the core volume of the fuel cell stack to be calculated[3].

8. Water flows

Water is produced not as steam, but as liquid in a PEMFC. A critical requirement of these cells is maintaining a high water content in the electrolyte to ensure high ionic conductivity. The ionic conductivity of the electrolyte is higher when the membrane is fully saturated, and this offers a low resistance to current flow and increases overall efficiency. Water content in the cell is determined by the balance of water or its transport during the reactive mode of operation.
Water management has a significant impact on cell performance, because at high current densities the mass transport issue associated with water formation and distribution limits cell output.
Without adequating water management, an imbalance will occur between water production and evaporation within the cell. Adverse effects include dilution of reactant gases by water steam, flooding of the electrodes, and dehydration of the solid polymer membrane. The adherence of the membrane to the electrode also will be adversely affected if dehydration occurs. Intimate contact between the electrodes and the electrolyte membrane is important because there is no free liquid electrolyte to form a conducting bridge. If more water is exhausted than produced, then it is important to humidify the incoming anode gas. If there is too much humidification, however, the electrode floods causes problems with diffusing the gas to the electrode. A smaller current, larger reactant flow, lower humidity, higher temperature, or lower pressure will result in a water deficit. A higher current, smaller reactant flow, higher humidity, lower temperature, or higher pressure will lead to a water surplus.
The balance of water emitted from the fuel cell reaction and that consumed by the reforming reaction is calculated from the mass balances of the reaction equations and the fuel utilisation factor. This neglects the water flow required to the fuel cell anode and cathode in order to prevent damage of the electrodes and polymeric membrane due to dehydration.

[3] Applicable only to the rectangular sandwich type construction.

9. Case study

Numerous PEMFC systems have been simulated, but to test the reliability of our program, we have simulated, in detail, the behaviour of a stack furnished from Ballard Power System [6]. This is based on a 5kW PEMFC fuel cell stack model MK-5E, composed of 36 cells; each cell has an active area of 232 cm^2.
For the system above mentioned, the polarization curve model of an individual cell at different temperatures, carried out from our program, can be seen in Figure 3. The coefficients of eq. (13), we have used to plot the graph below, were obtained from literature [6]. These curves are also plotted to aid specification of a suitable operating point. In Figure 4 is reported a system diagram including electric power and mass flows.

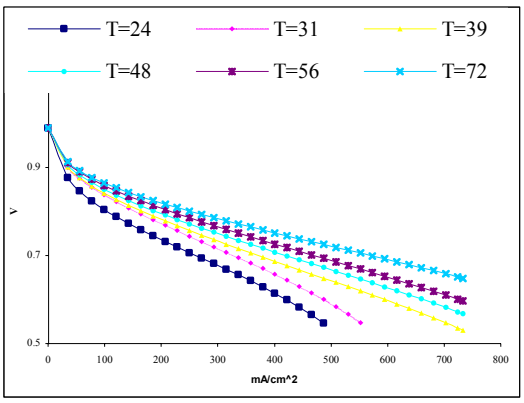


Figure 3: Polarization curves for different temperatures (T ≡ °C)

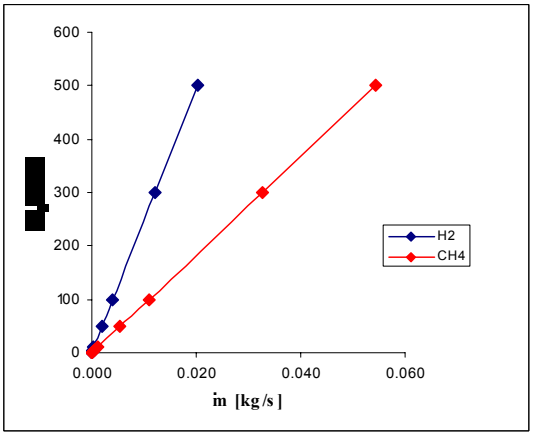


Figure 4: system diagram including main energy and mass flows

In Table 1 are listed some data, carried out from our program for some input data. The fuel used for this simulation is hydrogen and this one has been obtained from a natural gas with a reforming process. We have not taken into account the various losses due to pumps, compressors and so on.

Input data					
Reformer					
T [°C]	p [bar]	ε_R [%]			
950	3	74			
Fuel Cell					
T [°C]	p [bar]	V [Volt]	W_s [kW]	S [m^2]	N_T
24	3	0.99	5.0	232	36
ε_c [%]	R_i [W]	b [V]	m [V]	n [A^{-1}]	
95	1.353E-03	4.790E-02	1.2E-03	3.4E-05	
Output data					
$\dot{m}_{CH_4}$ [kg/s]	$\dot{m}_{H_2}$ [kg/s]	V_0 [V]	Q_{out} [kW]	W_R [kW]	
5.45E-04	2.03E-04	1.181	14.8	4.18	
ε_v [%]	ε_i [%]	ε_{el} [%]	ε_{hp} [%]	ΔG_T [kW]	
83.8	27.2	21.6	66.8	-23.1	
V_{st} [V]	I_{st} [A]	ln(K_p)	i [A/m^2]	I_T [A]	
35.64	147.68	0.255	6365.3	112.98	
$\dot{m}_a$ [kg/s]	ε_{max} [%]				
9.05E-04	94.3				

Table 1: Data carried out from the program for some input data

10. Conclusion

The program seems to be versatile and full of quantitative calculations that allow to study the behaviour of a single fuel cell or a single stack with a good reliability. Here has been taken into account only a hydrocarbon fuel such as the natural gas but the methodology can be simply applied to other fuels such as hydrogen, methanol, ethanol and so on. For sake of semplicity the model does not take into account that neither of the reforming reactions is perfect, since some natural gas and carbon monoxide make the FC through without reacting. By now we have not taken into account the various losses due to pumps, compressors and so on. Moreover the Kim et al. expression here adopted is limited to operation at constant stack pressure and so it does not include vital system parameters as stack temperature and relative humidity. However the immediate calculation of a lot of parameters with realistic values of FC performances can produce a good system to evaluate many quantities involved in the FC processes. The use of more complicated models

should produce likely better results but the parameters involved should be very numerous and sometimes difficult to estimate.
The Authors intend to validate this model trough laboratory experimentation on PEMFC by the Mechanics Department of Palermo University.

References

[1] J. Kim, S.M. Lee, S. Srinivasan, C.E. Chamberlin, *J. Electrochem. Soc.* 1995.

[2] J. O'M. Bockris and Amulya K. N. Reddy, *Modern Electrochemistry*, McGraw-Hill Book Company, 2002.

[3] F. Barbir, T. Gomez, *Int. J. Hydrogen Energy* 1997.

[4] *Fuel Cell Handbook 5th ed.*, U.S. Department of Energy, National Energy Laboratory, 2000.

[5] K. Kordesch, G. Simader, *Fuel Cell and Their Applications*, WILEY-VCH, Weinheim, 1996

[6] F. Laurencelle, R. Chahine, J. Hamelin, K. Agbossou, M. Fournier, T. K. Bose, A. Laperrière, *Characterization of a Ballard MK5-E Proton Exchange Membrane Fuel Cell Stack*, WILEY-VCH, Weinheim, 2001.

[7] L. Andreassi, S. Cordiner, *Valutazione delle prestazioni di un sistema di generazione di energia elettrica con celle a combustibile ad elettrolita polimerico mediante un modello di simulazione,* 56° Congresso nazionale ATI, Napoli, Settembre 2001 (in italian).

[8] N. Djilali, *Application of Computational Fluid Dynamics to Fuel Cell Modelling and Simulation*, Building the Hydrogen Economy, 11th Canadian Hydrogen Conference, Victoria, BC, Canada, June 2001.

[9] M. J. Khan and M. T. Iqbal, *Modeling and Analysis of Electrochemical, Thermal and Reactant Flow Dynamics for a PEM Fuel Cell System*, Fuel Cells, Wiley, May 2004.

[10] K. Hertwig, L. Martens, R. Karwoth, *"Mathematical Modelling and Simulation of Polymer Electrolyte Membrane Fuel Cells. Part I: Model Structures and Solving an Isothermal One-Cell Model"*, WILEY-VCH, Weinheim, 2002

[11] F. Lo Mastro, G. Blando, *Centrifugal water pump driven with a DC motor powered by an emulator of MK5-E proton exchange membrane fuel cell stacks*, 7TH Biennial Conference on Engineering Systems Design and Analysis, Manchester UK, Proceeding on CD, July 2004.

[12] P. Nguyen, *A Three-Dimensional Computational Model of PEM Fuel Cell with Serpentine Gas Channels*, M.A.Sc. Thesis, University of Victoria, September 2003.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

Modeling of Solar Radiation on Part Shaded Walls

M.Nikian and M. Naghashzadegan[1]
Department of Mechanical Engineering
University of Guilan, Rasht, Iran

Abstract

The program described in this research produces average solar radiation on walls after shading effects have been considered by numerical analysis methods. After extensive testing, it is hoped that, by integrating a version of this program into existing systems, a more realistic solar heat gain may be obtained for the site. As a result of this, more economical systems can be installed that will operate at more efficient levels benefiting both the user (capital and running costs) and the supplier (more competitive quotes). The program developed appeared to successfully deal with the majority of shading cases that are liable to be met in load estimating. The program is quick and reasonably accurate (to within about 2% of hand calculation) with as few as 100 data points.
Keywords: Solar radiation, cooling load calculation, shaded wall

Nomenclature

AZ = solar azimuth
AL = solar altitude
Dpx = x-co-ord of a point to be analyzed
Dpy = y-co-ord of a point to be analyzed
Dpz = z-co-ord of a point to be analyzed
SVX = solar vector x component
SVY = solar vector y component
SVZ = solar vector z component
SOX = shading wall origin point x
SOY = shading wall origin point y
SOZ = shading wall origin point z
WAHX = horizontal shading wall length (x direction)
WAHY = horizontal shading wall length (y direction)
WAV = vertical height of a shading wall
ITH = Total intensity on horizontal surfaces (w/sqm)
IDH = Direct intensity on horizontal surfaces (w/sqm)
Idh = Diffuse intensity on horizontal surfaces (w/sqm)
IDV = Direct intensity on vertical surfaces (w/sqm)
ITV = Total intensity on vertical surfaces (w/sqm)

Introduction

To calculate zone loads, all load components must be considered separately as internal or external loads. A typical set of loads might be: Solar gain, Glass transmission, Wall transmission, Roof transmission, Lighting, Other electrical, People and Cooling infiltration. Each of these may be further sub-divided into sensible and latent to determine the overall cooling or heating air supply required. The magnitude of the components varies and peak at different times. For outside wall zones, the solar gain is often a very significant component (for instance up to 50% [1]), depending on the position of the sun and the size of the glazing. The large importance of the solar gain component is rarely matched by the sophistication of the calculations used. Thus, zone peaks and subsequent sizing may be highly influenced by errors generated within calculations.

The estimation of heating and cooling loads on a building prior to the installation of an air conditioning system is both complex and liable to large errors. Most systems are now sized by

[1] Corresponding author: Tel/Fax: (+98) 131- 3232204 , E-Mail: naghash@guilan.ac.ir

commercial load estimating computer programs. Large errors can be sliced from solar gain estimations if shading is included. Load estimating programs for determining the size of air conditioning programs take into account many heat gain sources. One of the most significant external gains arises due to solar radiation on walls and windows. The calculation of solar radiation is often unrealistically high due to the lack of consideration given to shading on the external walls.

Most commercial load estimating programs were found to be without building shading equations. Two programs investigated, (CIBSE Heat Gains [2] and Carrier HAP [3]) were found to identify shading associated with external wall features.

A third program was investigated in detail is an advanced load estimating program written in 1976 in the USA [4]. The program was devised by the National Bureau of Standards at the Centre for Building Technology.

Objectives:

To investigate an effective method for the calculation of solar radiation on a building walls the position and orientation of other interfering walls and building. The output should be calculated with a view to using it as an input to additional program modules (for instance - to investigate dynamic heat transfer through the building surface).

The method should be fast, accurate and verifiable. It should also be easy to use but flexible. The project should relate to programs presently in use and how the new programs might integrate into existing methods.

Solar Position Definitions

The sun position is given by two angles; the solar azimuth and the solar altitude. Their definitions are as follows:

a: The solar altitude (al)

The angle a direct ray from the sun makes with the horizontal at a particular place on the surface of the earth (Figure 1).

$$\sin(a_1) = \sin(D) \times \sin(LAT) + \cos(D) \times \cos(LAT) \times \cos(h) \quad (1)$$

b: The solar azimuth (z)

This is the angle the horizontal component of a direct ray from the sun makes with the true North-South axis. It is expressed as an angular displacement through 360 degrees from true North (in the clockwise direction) (Figure 2).

To calculate both of the above angles, it is necessary to know the sun's position relative to the plane of rotation of the Earth (declination) and the position of the site on the surface of the Earth (latitude). Also the local time must be used for the calculation of the 'sun-time'. This eliminates the need for the longitude of the site to be entered. The following definitions list this required information:

$$\tan(z) = \frac{\sin(h)}{\sin(LAT) \times \cos(h) - \cos(LAT) \times \tan(D)} \quad (2)$$

c: Declination (d)

This is the angular displacement of the sun from the plane of rotation of the Earth's equator. The value of the declination will vary throughout the year between -23.5° and +23.5° because the Earth is tilted at an angle of about 23.5° to the axis of the plane in which it orbits the sun. Figure 3 shows the relationship.

$$D = 23.47 \sin\left(360 \times (284 + N)/365\right) \quad (3)$$

Where N is the day number (January 1st=1)

d: Latitude (LAT)

An angular displacement above or below the plane of the equator measured from the centre of the earth, gives the latitude of a site (shown in Figure 4).

e: Sun time (T)

This is the time in hours before or after noon.

f: Hour angle (h)

The angular displacement of the sun from noon.

$$h = (360/24) \times T \quad (4)$$

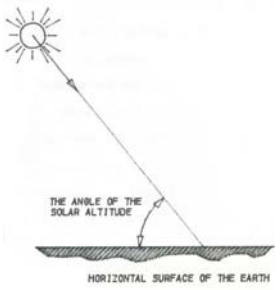


Figure 1: Soalr altitude

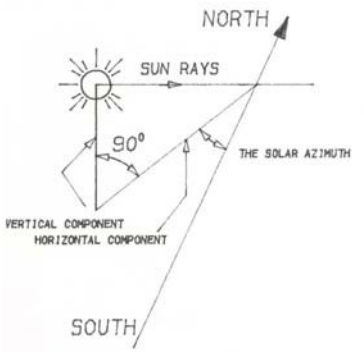


Figure 2: Solar Azimuth

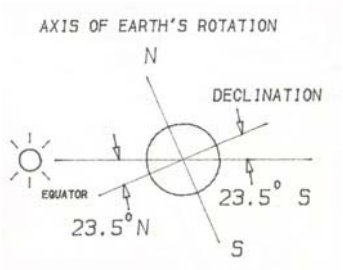


Figure 3: Declination

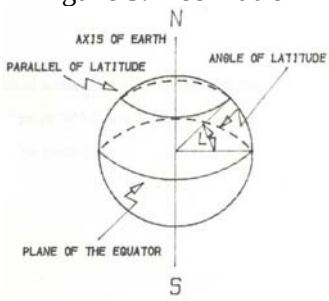


Fig. 4: Latitude

Solar Radiation

1- Direct Radiation

The intensity of direct radiation on a vertical surface is easily calculated if the beam radiation 'N' is known. For a wall-solar angle of 'WAZ' and a solar altitude 'al' the intensity of the direct vertical component is given by:

$$IDV = N \times \cos(a_1) \times \cos(WAZ) \qquad (5)$$

2- Diffuse radiation

Direct radiation entering the Earth's atmosphere is subject to scattering to create 'sky radiation' or 'diffuse radiation'.

The processes by which this occurs can be split into four categories:

i) Radiant energy scattered by atmospheric molecules of ideal gas (eg. nitrogen, oxygen)
ii) Scattering due to presence of water vapor
iii) Selective absorption of water vapor
iv) Scattering by dust particles

Sky radiation can't be assigned a specific direction (and hence no shadows are cast by it). The intensity of sky radiation is usually much less than that for direct radiation but cannot be ignored. The quantity of sky radiation varies with atmosphere's variation of gas composition, water vapor content, and dust content. It also varies as the solar altitude changes.

3- Shading by Walls and Buildings

The main purpose of the program is to provide a reliable and fairly quick method of building shading analysis. The three dimensional geometry is fairly simple but difficult to generalize the formula into one simple case. To establish whether shading of a point (on the wall to be analyzed) occurs, the geometry of the site and surrounding buildings require to be known. The definition of the walls and buildings on the site are covered in detail in the program section. However, the walls may be assumed to be rectangular and perpendicular to the ground and to be completely opaque.

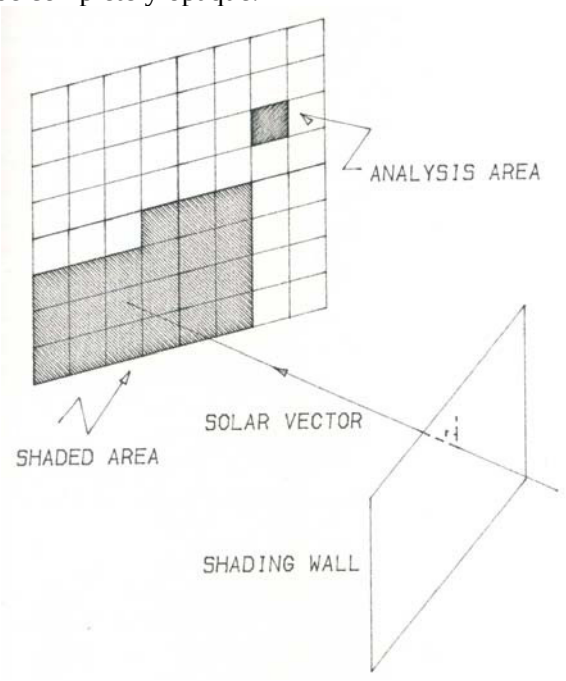


Figure 5: Solar vector and shading wall intersection

Derivation of the Shading Equations

The derivation of the following formulae is complicated somewhat by the problems of 1/0 errors (caused by tan(A/B) when B=0; l/cos(A), l/sin(A) when A=0). The formulae have to be rearranged to cater for such eventualities and a test routine at the start of this section would then be required to ensure the correct equations are used. The diagram (Fig. 5) shows the typical situation and the definition of the terms is given below:

General Equation:

$$Dpx+k_1(SVX)=SOX+k_2(WAHX) \quad (6)$$

$$Dpy+k_1(SVY)=SOY+k_2(WAHY) \quad (7)$$

$$Dpz+k_1(SVZ)=SOZ+k_3(WAV) \quad (8)$$

Where k_1, k_2, k_3 are three unknown constants to be found

Solution to General Equation

Rearranging general equations (6), (7) and (8):

$$k_2(SVX)=(COXED)+k_1(WAHX) \quad (9)$$

$$k_2(SVY)=(SOY-Dpy)+k_1(WAHY) \quad (10)$$

$$k_2(SVZ)=(SOZ-Dpz)+k_3(WAV) \quad (11)$$

Multiply equation (9) by (SVY/SVX):

$$k_2(SVY)=(SVY/SVX)\times(COXED) + k_1(SVY/SVX)\times(WAHX) \quad (12)$$

Subtracting by equation (10):

$$0=(SVY/SVX)\times(COXED)-(SOY-Dpy) + k_1(((SVY/SVX)\times WAHX)-WAHY) \quad (13)$$

So,

$$k_1=\frac{[((SOY-Dpy)-(SVY/SVX)\times(COXED))]}{[((SVY/SVX)\times WAHX)-WAHY]} \quad (14)$$

And from equation (10) we have:

$$k_2=\frac{[(SOY-Dpy)-k_1(WAHY)]}{SVY} \quad (15)$$

Also using equation (11) gives:

$$k_3=\frac{[(SVZ/SVY)((SOY-Dpy)+k_1(WAHY))-(SOZ-Dpz)]}{WAV} \quad (16)$$

Subsisting for k_1:

$$k_3=\frac{WAHY\times((SOY-Dpy)-(SVY/SVX))\times(COXED)}{WAV\times((SVY/SVX)\times(WAHX-WAHY))} + \frac{[(Dpz-SOZ)+(SVY/SVX)(SOY-Dpy)]}{WAV} \quad (17)$$

Spatial Cases:

This is a list of some possible problems to be considered when the general equation is to be solved. All those listed below indicate 1/0 errors may arise during calculation:

1. WAHX=0: wall is aligned N-S or S-N
2. WAHY=0: wall is aligned E-W or W-E
3. SVX=0: sun is due south (midday)
4. SVY=0: sun is due east or due west
5. SVZ=0,<0: sunrise/sunset and during night

Solution for Special Equations:

Special cases require modification to the general solution. These will be dealt with in order:

Case1 (WAHX=0)

$$k_1=\frac{[(SVY/SVX)\times(COXED)+(Dpy-SOY)]}{WAHY} \quad (18)$$

$$k_2=\frac{(COXED)}{SVX} \quad (19)$$

Hence:

$$k_3=\frac{[(SVZ/SVY)(COXED)+(Dpz-SOZ)]}{WAV} \quad (20)$$

Case 2 (WAHY=0)

As long as the wall vector WAHY is not zero, the expressions derived for the general equation can be used with SVY=0. This produces:

$$k_1=\frac{[(SVY/SVX)(SOY-Dpy)+(Dpx-SOX)]}{WAH} \quad (21)$$

$$k_2=\frac{(SOY-Dpy)}{SVY} \quad (22)$$

$$k_3=\frac{[(SVZ/SVY)(SOY-Dpy)+(Dpz-SOZ)]}{WAV} \quad (23)$$

Case 3 (SVY=0)

As long as the wall vector WAHY is not zero, the expression derived for the general equation can be used with SVY=0. This produces:

$$k_1=\frac{(Dpy-SOY)}{WAHY} \quad (24)$$

$$k_2=\frac{[(COXED)+(WAHX/WAHY)(Dpy-SOY)]}{SVX} \quad (25)$$

$$k_3 = \frac{(SVZ \times WAV)[(COXED) + (WAHX / WAHY)(Dpy - SOY)]}{SVX} - \frac{(SOZ - Dpz)}{WAV} \quad (26)$$

Case 4 (SVX=0)

Rearranging the general equations and assuming that SVY is not zero:

$$k_1 = \frac{(Dpx - SOX)}{WAHX} \quad (27)$$

$$k_2 = \frac{[(SOY - Dpy) + (WAHY/WAHX)(Dpx - SOX)]}{SVY} \quad (28)$$

$$k_3 = \frac{SVZ \times [(SOY - Dpy) + (WAHY/WAHX)(Dpx - SOX)]}{SVY \times WAV} - \frac{(SOZ - Dpz)}{WAV} \quad (29)$$

Case 5 (SVZ<0 or SVZ=0)

This can easily be detected before any calculation is carried out and the physical interpretation of this is that the sun has not risen. The solar data for this case is thus considered to be zero and no further shading calculation are therefore required.

Calculation of SVX, SVY, SVZ

The calculation of these vectors may be easily achieved by the use of the altitude and azimuth angles as derived earlier:

$$SVX = \cos(a_1)\sin(az)$$
$$SVY = \cos(a_1)\cos(az) \quad (36)$$
$$SVZ = \sin(a_1)$$

Solar Data Generation

This method uses clear sky solar data generated by sinusoidal equations that have been fitted to experimental data. Assuming that the solar angles are already known:

D = Declination, *L* = Latitude, *A* = Solar altitude, *Z* = Solar azimuth, *S* = Face orientation of window analyzed; Then the sun normal intensity (*N*) w/sqm at each hour is:

$$N = 1074.16\times\sin(A) + 1980.060\times\sin(3A) + 70.1766\times\sin(5A) + 30.3902\times\sin(7A) + 13.3842\times\sin(9A) + 5.59234\times\sin(11A) + 2.93048\times\sin(13A) + 0.606472\times\sin(15A) \quad (37)$$

It is then necessary to correct the intensities by applying an altitude correction factor (K_a) for sites with an elevation 300m or greater above sea level:

$$K_a = 1.02 + 0.00002\times\text{elevation} + 0.00005\times\text{elevation}\times(1/\sin(\text{A})) \quad (38)$$

$$Idh = 121.649\times\sin(A) + 14.7575\times\sin(3A) + 7.72576\times\sin(5A) + 3.47353\times\sin(7A) + 2.22222\times\sin(9A) + 0.52539\times\sin(11A) + 0.52164\times\sin(13A) + 0.1311\times\sin(15A) + 0.7687\times\sin(15A) \quad (39)$$

For clear sky conditions:

I=1 Overall radiation factor, K_c=0.95 Direct radiation factor, K_r=0.2 Ground reflection factor, *c*=0 Cloudiness, fc_1=l Cloudiness factor

$$ITH = Ka \times I \times (K_c \times fc_1 \times IDH \times Idh) \quad (40)$$

$$IDV = N \times \cos(A) \times \cos(Z - S) \quad (41)$$

$$ITV = K_a \times I \times (K_c \times fc_1 \times IDV + 0.5 \times Idh + 0.5 \times K_r \times ITH) \quad (42)$$

Window is specified on, the window will receive only diffuse radiation and hence:

IDV = 0, IDH = 0

Numerical Analysis

For a general case solution to all the possible geometrical problems posed by shading, analytical techniques would be cumbersome and very complex, and not necessarily quicker or more accurate. Analytical techniques could be used to find the shading boundaries of the problem; numerical analysis is especially useful for calculating the area that is shaded.

Numerical techniques require the splitting up of an area to be analyzed into smaller areas. Each area is assigned a central point where the equations for that area are evaluated. It is assumed that the conditions at this point are then valid for the rest of the area. Thus the problem is broken down into discrete point analysis rather than the calculation of a continuum. This is more straightforward, and is relatively easy to convert into a computing sequence.

In previous part, , a set of equations were derived to establish whether shading of a particular point *(Dpx, Dpy, Dpz)* occurs due to

another arbitrarily positioned wall at a certain time. It can be seen that it is fairly easy to incorporate the testing of a series of data points into the analysis of a single larger area. Thus the style of the program emerges: The testing of these shading equations on sets of data point co-ordinates throughout the building will establish the fractions of the walls that are shaded.

Results and Discussion

This section gives an example of which aims to demonstrate the advantages of the program. This is shown as plan view of the site, a listing of the results and graphs of the solar radiation in watts per square meter on the windows analyzed. The building have been analyzed for two time intervals, January 800hrs - 1600hrs and June 800hrs - 1600hrs. This helps show the annual variations of sun position and the consequent variations in the shading patterns. The significance of the shading is very evident.

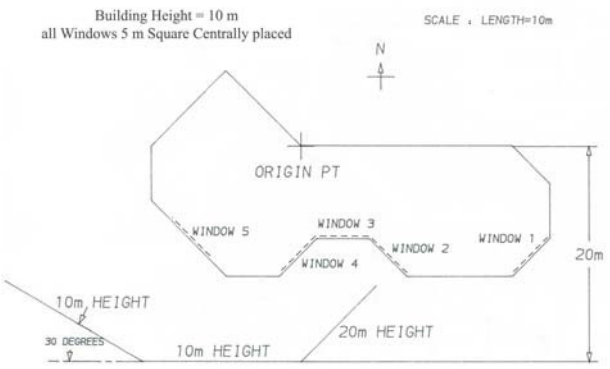


Figure 6: plan view of example

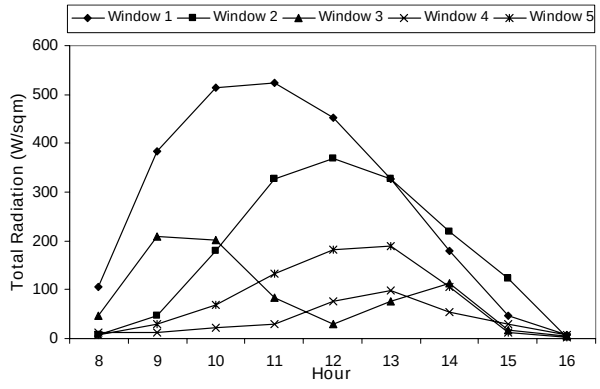


Figure 8: Total Radiation for 5 Windows in Example Building
Time Interval of January 800hrs – 1600 hrs

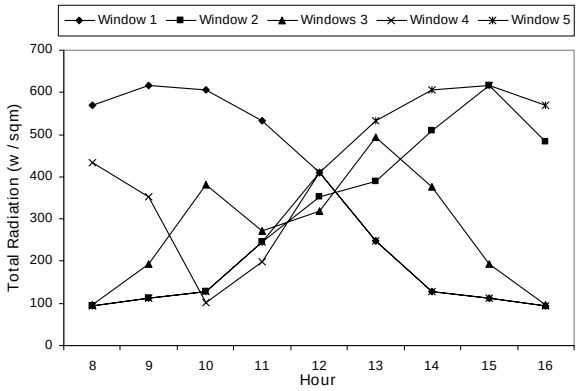


Figure 8: Total Radiation for 5 Windows in Example Building
Time Interval of June 800hrs – 1600 hrs

Conclusion

The overall objective of this research was to stress how important the effects of shading are on the incident solar radiation on the building (and consequently the cooling load on the air-conditioning equipment).

The program described in this research produces average solar radiation on walls after shading effects have been considered by numerical analysis methods. After extensive testing, it is hoped that, by integrating a version of this program into existing systems, a more realistic solar heat gain may be obtained for the site. As a result of this, more economical systems can be installed that will operate at more efficient levels benefiting both the user (capital and running costs) and the supplier (more competitive quotes).

The program developed appeared to successfully deal with the majority of shading cases that are liable to be met in load estimating. The program is quick and reasonably accurate (to within about 2% of hand calculations) with as few as 100 data points. The program apparently indicates the shading equations are correct together with the methodology behind their use.

References:

1- Calculation done using Carrier E20-II HAP (version 1.1), August 1989.
2- CIBSE Heat Gains (version 2.13) program.
3- Carrier HAP (version 1.1) program.
4- Kasuda, T NBSLD, “The computer program for heating and cooling loads”, 1976.

Shading Radiation Data for Window 1 (100 Data Points)						
Month	Time (hr)	ITV (w/sqm)	IDV (w/sqm)	%Direct Rad.	Total (w/sqm)	Total Watts
Jan	8:00	105.71	103.41	100	105.71	2643.1
Jan	9:00	384.52	373.09	100	384.52	9614.2
Jan	10:00	515.07	493.91	100	515.07	12878.4
Jan	11:00	524.28	493.67	100	524.28	13108.8
Jan	12:00	451.73	413.97	100	451.73	11294.8
Jan	13:00	327.00	286.01	100	327.00	8176.0
Jan	14:00	178.76	139.89	100	178.76	4469.5
Jan	15:00	45.94	16.70	100	45.94	1148.8
Jan	16:00	7.47	0.00	0	7.47	186.8

Table 1: Calculation Results for Window 1 in Example Building in January

Shading Radiation Data for Window 1 (100 Data Points)						
Month	Time (hr)	ITV (w/sqm)	IDV (w/sqm)	%Direct Rad.	Total (w/sqm)	Total Watts
Jun	8:00	569.17	499.74	100	569.17	14231.0
Jun	9:00	617.58	531.06	100	617.58	15441.5
Jun	10:00	604.68	503.12	100	604.68	15119.0
Jun	11:00	532.54	417.95	100	532.54	13315.4
Jun	12:00	408.94	284.46	100	408.94	10224.8
Jun	13:00	246.88	117.25	100	246.88	6172.7
Jun	14:00	126.72	0.00	0	126.72	3168.4
Jun	15:00	113.07	0.00	0	113.07	2827.2
Jun	16:00	94.41	0.00	0	94.41	2360.7

Table 2: Calculation Results for Window 1 in Example Building in June

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

OPTIMIZATION OF THE HEN OF A REFORMING SYSTEM

Mads Pagh Nielsen[1], Anders Korsgaard[2] and Søren Knudsen Kær[3]

[1] *Assistant Professor at Institute of Energy Technology, Thermal Energy Systems, Aalborg University (AAU, IET),*
[2] *Ph.D. Student at Institute of Energy Technology, Thermal Energy Systems, Aalborg University (AAU, IET),*
[3] *Associate Professor at Institute of Energy Technology, FACE, Aalborg University (AAU, IET).*

ABSTRACT

Proton Exchange Membrane (PEM) based combined heat and power production systems are highly integrated energy systems. They may include a hydrogen production system and fuel cell stacks along with post combustion units optionally coupled with gas turbines. The considered system is based on a natural gas steam reformer along with gas purification reactors to generate clean hydrogen suited for a PEM stack. The temperatures in the various reactors in the fuel processing system vary from around 1000°C to the stack temperature at 80°C. Furthermore, external heating must be supplied to the endothermic steam reforming reaction and steam must be generated. The dependence of the temperature profiles on conversion in shift reactors for gas purification is also significant. The optimum heat integration in the system is thus imperative in order to minimize the need for hot and cold utilities. A rigorous 1D stationary numerical system model was used and process integration techniques for optimizing the heat exchanger network for the reforming unit are proposed. Objective is to minimize the system cost.

Keywords: Fuel cells; Steam Reforming; Heat Exchanger Network (HEN) Synthesis; MINLP.

NOMENCLATURE

c_i	Molar concentration [mol/m^3]
C,β	Area cost coefficient and exponent
CCU/HU	Cost of cold and hot utility
C_{is}	Concentration in solid [mol/m^3]
$c_{p,g}$	Spec. heat capacity of gas [J/(kg·K)]
d_p	Equivalent particle diameter [m]
$dtcu_{i/j}$	Approach temp. for match i and utility [K]
$D_{t,i/e}$	Internal-/External tube diameter [m]
$D_{eff,i}$	Effective diffusion coefficient [cm²/s]
D_{er}	Eff. radial diffusivity [m_{fluid}/($m_{reactor}$·s)]
f	Friction factor [-]
F	Heat capacity flow rate
F_i	Molar rate of specie *i* [kmol/h]
$F_{total,i}$	Total molar flow rate at inlet [kmol/h]
ΔH_i	Enthalpy of reaction [kJ/mol]
LMTD	Logarithmic mean temperature difference [K]
p	Total pressure [bar]
p_0	Total pressure at inlet [bar]
p_i	Partial pressure of specie *i* [bar]
Pr	Prandtl number [-]
qcu_i/hu_j	Heat exchanged between stream and utility.
q_{ijk}	Heat exchanged between stream (i,j) in stage k.
r	Radial coordinate [m]
r_j	Reaction rate [kgmol/h/kg cat.]
r_w	Radius at wall [m]
R	Universal Gas Constant [J/(mol·K)]
Re	Reynolds number [-]
R_i	Rate for specie [kgmol/h/kg cat.]
T	Gas temperature [K]
TIN/OUT	Inlet/outlet temperature of stream [K]
T_0	Inlet gas temperature [K]
T_{is}	Temperature within the solid particle [K]
T_{wall}	Wall temperature [K]
ΔT_{ijk}	Approach temperature match (i,j) loc. k [K]
u_s	Superficial velocity [m/s]
z	Axial coordinate [m]
zcu_i/hu_j	Binary variable denoting utility match
z_{ijk}	Binary variable denoting match (i,j) at stage k
α_w	Wall heat transfer coefficient [J/m²·h·K]
ε	Void fraction of packing, [m^3_{fluid}/m^3_{bed}]
λ_{eff}	Effective radial conductivity [J/(m·s·K)]
$\lambda_{eff,0}$	Static eff. radial conductivity [J/(m·s·K)]
ρ_B	Bed density (incl. porosity) [kg/m^3]
ρ_g	Gas density [kg/m^3]
ρ_s	Density of the solid [kg/m^3]
$\eta_{i/j}$	Effectiveness of the catalyst, j~r_j&i~R_i [-]
ξ	Pellet coordinate in active area, [m]
Ω	Cross-sectional pipe area [m²]

[1] Corresponding author, mpn@iet.aau.dk

INTRODUCTION

PEM fuel cells are electrochemical devices, which efficiently convert energy in hydrogen directly into electricity without combustion and with no moving parts. The process is the opposite of electrolysis and can be compared with the process in a battery. The following figure illustrates the principle of a single fuel cell:

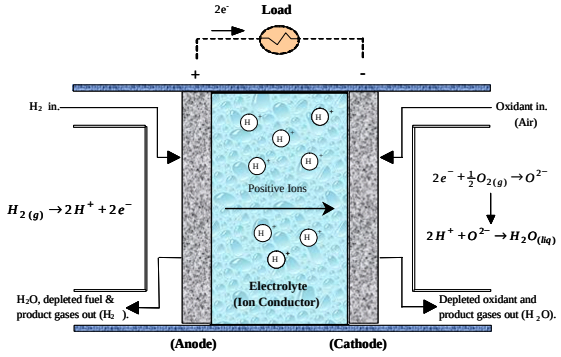


Fig. 1: Single hydrogen fuel cell.

The polymer electrolyte separates the reactants and prevents electronic conduction. The electrolyte allows protons to pass through via electro-osmotic drag. Electrons feed from the reaction then pass through an external loop to supply the given load and take part in the cathode reactions.

PEM fuel cells have a potential of approximately 0.6-0.7 volts, which means that serial stacking is necessary in order to get the desired voltage. The photo below shows a recent 800W Pt-Rt catalyst - 50 cell PEM stack developed at FACE AAU, IET:

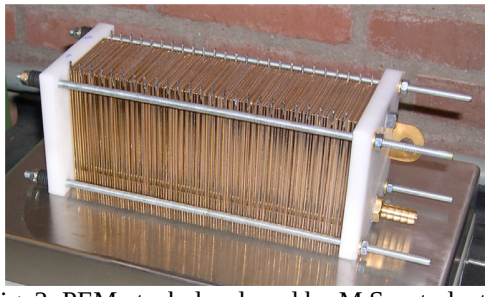


Fig. 2: PEM-stack developed by M.Sc. students.

Hydrogen is not available as conventional fossil fuels like natural gas, oil and coal so it must be produced either from renewable energy driven electrolysis or from fuel processing of hydrocarbons. This paper deals with hydrogen production by steam reforming (SR) of natural gas (NG).

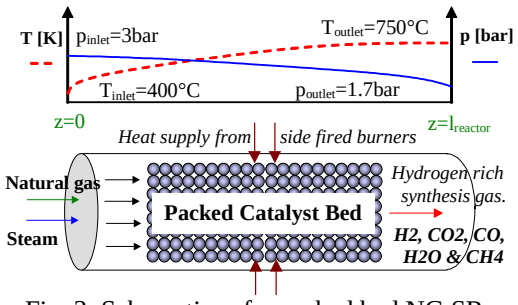


Fig. 3. Schematics of a packed bed NG SR.

Steam reforming in catalytic packed beds (fig. 3) has been dealt with extensively. The main advantage compared to other fuel processing technologies is the high efficiency. The catalyst particles are illustrated as spheres in fig. 2 but can have various other geometrical shapes. The shape depends on a tradeoff between pressure losses through the bed and the active surface area.

Practical and theoretical issues about steam reforming were described by Rostrup-Nielsen [1] in detail. Froment & Xu [2a]+[2b] developed and validated a generalized Langmuir-Hinshelwood type kinetic model of the process (assuming that all molecules at the catalyst surface are adsorbed). This model has been widely accepted. Through this work, it was recognized that mass transfer limitations within catalyst pellets are significant for this process, which is strongly diffusion controlled.

Only a thin layer of catalyst surface contributes to the process. Levent et al. [3] investigated this phenomenon further using a simplified approach for spherical catalyst particles.

Rajesh et al. [4] developed a steam reformer model based on the work of Froment & Xu [2a]+[2b]. They used a genetic algorithm for optimizing plant economy of large-scale hydrogen production plants.

Godat et. al. [19] used a process integration technique to make a preliminary system analysis of a

PEM-steam reforming system. They proved that the theoretical efficiency could be improved from 35% to 49% using appropriate process integration.

SYSTEM DESCRIPTION

A simplified PEM fuel cell steam reforming system is illustrated on fig. 4. The temperature levels of the different processes are illustrated as well. The system involves several reactor steps due to gas impurity issues [8]. In the following, the individual steps needed in producing hydrogen for PEM fuel cells will be described.

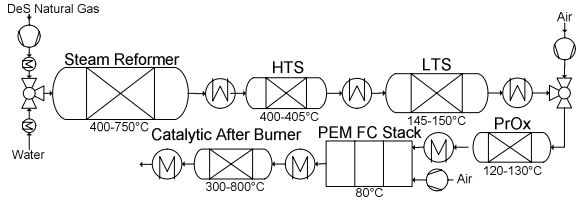


Fig. 4. SR-system with H_2-purification system.

Desulphurization
Natural gas desulphurization is necessary due to the sulphur content in most natural gas sources. Sulphur compounds poisons to the nickel catalyst used in steam reformers and is also severely degrades the efficiency of PEM fuel cells. The usual DeS-process is hydrodesulphurization (convert the sulphur into H_2S followed by a ZnO bed to convert the H_2S into ZnS(s) and steam [9]).

CO-poisoning of PEM-stacks
Carbon monoxide (CO) is a severe poison in PEM fuel cell stacks since it adsorbs to the electrode. The efficiency is strongly degraded if the CO content in the syngas exceeds 10ppm [8]. The effect is time dependant [10]. The process, however, is reversible, which means that the PEM stack is fully recovered again after some time with operation on pure hydrogen. Other methods for faster recovery are under development. One method is to add a little oxygen to the anode gas. This minimizes the problem with CO but leads to more advanced heat management in the fuel cell stack due to the exothermic nature of the CO-oxidation.

Water Shift Converters
It is necessary with two steps (LTS & HTS) to get rid of CO. Typically water gas shift converters are used in two temperature steps to minimize catalyst volume. Steam reacts with CO and forms CO_2 and H_2 and thus contributes to hydrogen production.

Preferential Oxidation
Subsequent to the shift conversion steps the carbon monoxide content is still 1-2%, which means that further purification is necessary. This is done by adding a small fraction of air to the gas in a catalytic reactor. This process is strongly exothermic so it is usually done in several steps with intermediate cooling to be able to control the process. In the conversion process a small amount of the hydrogen is lost due to reaction with oxygen forming water. 0.5-1% of the H_2 is lost in this process.

STEAM REFORMER MODEL

For simplicity, it was chosen to use the 1D-model developed by Froment & Xu [2a]+[2b].

Reaction kinetics
The overall reaction scheme for the process including enthalpies of reaction is summarized as:

$$CH_4 + H_2O \rightarrow CO + 3H_2, \Delta H = 224.0\ kJ/kmol \quad (1)$$

$$CO + H_2O \rightarrow CO_2 + H_2, \Delta H = -37.3\ kJ/kmol \quad (2)$$

$$CH_4 + 2H_2O \rightarrow CO_2 + 4H_2, \Delta H = 186.7\ kJ/kmol \quad (3)$$

Rate expressions expressing the kinetic rate of disappearance or production of species as function of the partial pressures (i.e. expressions representing reaction kinetics) can be found in the original papers [2a] & [2b].

Conservation equations
Since only reaction (1) and (2) are linearly independent it is possible to express the overall conversion in terms of two conversion species. Other components can be found in terms of the molar rates and the conversion of these. It is chosen to select CH_4 and CO_2 as key constituents. Plug flow condition (linear velocity profile) is assumed.

The species balance along the reactor coordinate may be written:

$$\frac{dF_{CH_4}}{dz} = \Omega\rho_B\eta_{CH_4}R_{CH_4} = \Omega\rho_B(-\eta_1 r_1 - \eta_3 r_3) \qquad (4)$$

$$\frac{dF_{CO_2}}{dz} = \Omega\rho_B\eta_{CO_2}R_{CO_2} = \Omega\rho_B(\eta_2 r_2 + \eta_3 r_3) \qquad (5)$$

The stationary energy balance of the reactor yields:

$$\frac{dT}{dz} = \frac{\left[\rho_B\left(-\Delta H_1\eta_1 r_1 - \Delta H_2\eta_2 r_2 - \Delta H_3\eta_3 r_3 - \frac{4U(T - T_{wall})}{D_{Tube,i}}\right)\right]}{\rho_g c_{p,g} u_s} \qquad (6)$$

The momentum equation expressing the pressure loss through the packed bed is:

$$\frac{dp_t}{dz} = -f\frac{\rho_g {u_s}^2}{d_{particle}} \qquad (7)$$

Ergun's equation [11] was used to calculate the friction factor and the equivalent particle diameter is defined as the diameter of a sphere with the same external surface area per unit volume of the catalyst particle, see VDI Wärmeatlas, page Gg 3 (1974). Initial values for (4), (5), (6) & (7) are zero conversion and inlet temperature and pressure. A heat transfer model was implemented to calculate U and T_{wall}. Thermodynamic properties were calculated using standard ideal gas mixture rules. Mixture heat conductivity and viscosity were found using Wilke's method.

Calculation of effectiveness factors

The effectiveness factors in (4), (5) & (6) account for intra particle resistance and were calculated assuming a simplified slab geometry of the catalyst pellets. The factors express the ratio between the actual rates due to mass flow resistance in the pellets and the ideal rates based on the catalytic surface. The pellet geometry is illustrated on fig. 5

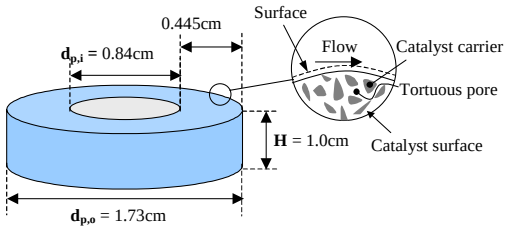


Fig. 5. Pellet Geometry.

For the slab geometry the concentration profiles in the catalyst is evaluated at each axial point by the following type boundary value problem (BVP) differential equation equating the sum of in- and outlet fluxes to the species production:

$$\frac{D_{eff}}{\xi^2}\frac{d}{d\xi}\left(\xi^2\frac{dC_{is}}{d\xi}\right) = \rho_s R_{is}(C_{is}, T_{is}) \qquad (8)$$

It can be assumed that the pellets have same temperature as the bulk flow so that the mass balance in (8) is sufficient to describe the intra particle flow. Initial values is zero gradient at ξ=0 and concentrations equal to bulk conditions at ξ=thick$_{cat}$.

The BVP was solved using orthogonal collocation on finite elements converting the differential equations into a set of non-linear algebraic equations. The solution was approximated using LaGrange polynomials. The collocation points were chosen as the roots of the Legendre polynomial. See [14] for a detailed description of this method.

Average molar fractions produced by the steady state steam reformer model are shown on fig. 6:

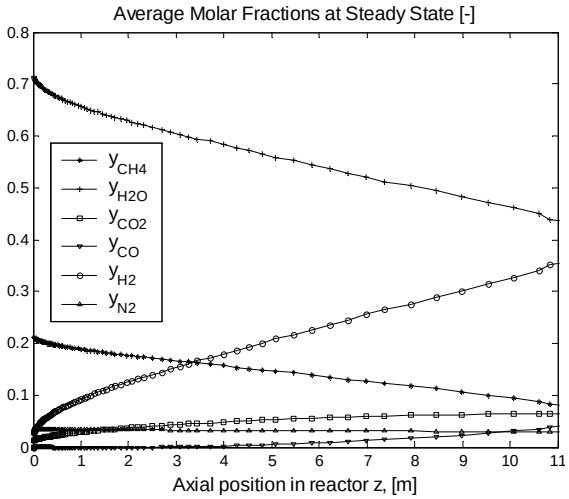


Fig. 6. Species profiles through reactor.

The results correlate very well with simulation results publicized in the literature elsewhere [2a+2b], [4].

Thermodynamic properties and heat transfer
Thermodynamic mixture properties were calculated using standard ideal gas mixture rules. Mixture transport properties such as thermal conductivity and viscosity were found respectively using Wassiljewa's method as described in [12] based on normal boiling points and Wilke's method as described in [13]:
Mixture data for the transport properties (i.e. thermal conductivity and viscosity) was found using Wilke's method and the regression data for the transport properties of the individual species. This was done using the interaction coefficients as:

$$\mu_{mix} = \sum_{i=1}^{NS} \frac{x_i \mu_i}{\sum_{j=1}^{NS} x_j \phi_{ij}} \quad \text{and} \quad k_{mix} = \sum_{i=1}^{NS} \frac{x_i k_i}{\sum_{j=1}^{NS} x_j \phi_{ij}} \qquad (9)$$

Basically, the following procedure can be used in calculating the interaction coefficients in for the mixture viscosity (10):

$$\phi_{ij} = \varphi_{ij} = \frac{\left[1 + \sqrt{\frac{\mu_i}{\mu_j}} \cdot \left(\frac{M_j}{M_i}\right)^{\frac{1}{4}}\right]^2}{\sqrt{8} \cdot \left[\sqrt{1 + \frac{M_i}{M_j}}\right]} \qquad (10)$$

Effective diffusivities for each gas in the mixture have been calculated using the method described in [12] as a geometric mean of the Knudsen diffusion and binary molecular mixture diffusivities.

In addition [12] gives the wall heat transfer coefficient and the effective radial conductivity in the fixed bed based on the work of De Wasch and Froment (1972) as follows [11]:

$$\alpha_w = \frac{8.964 \lambda_{eff,0}}{D_{t,i}^{4/3}} + \frac{0.512 \lambda_g D_{t,i} \mathrm{Re}_{(d_p)} \mathrm{Pr}^{\frac{1}{3}}}{d_p} \qquad (11)$$

$$\lambda_{eff} = \lambda_{eff,0} + \frac{0.111 \lambda_g \mathrm{Re}_{(dp)} \mathrm{Pr}^{\frac{1}{e}}}{1 + 46\left(\frac{d_p}{D_{t,i}}\right)^2} \qquad (12)$$

Where:

$$\lambda_{eff,0} = \varepsilon\left(\lambda_g + 0.95 a \cdot d_p\right) + \frac{0.95(1-\varepsilon)}{2.23 + \left[10\lambda_g + b \cdot d_p\right]^{-1}} \qquad (13)$$

and

$$a = \frac{0.1952\left(\frac{T}{100}\right)^3}{1.0 + \left[\frac{0.25\varepsilon}{2(1-\varepsilon)}\right]} \quad \& \quad b = \frac{0.1952\left(\frac{T}{100}\right)^3}{3} \qquad (14)$$

The heat transfer model accounts for radiation and conduction between voids in the fluid phase and transport between neighboring particles in the solid phase.

SHIFT REACTORS & PROX

Fundamentally, the shifts- and PROX reactors are modeled with similar assumptions as done in the steam reformer model. The reaction kinetics and catalyst properties used were taken from [15]. The PROX can be modeled using the kinetics in [16].

MODELING OF PEM STACK

The PEM stack was modeled simplified calculating the polarization curve expressing the correlation between the stack potential and the current density (fig. 7):

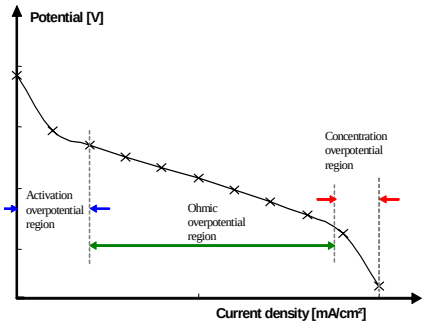


Fig. 7. Typical polarization curve for PEM-stack.

Using the fuel cell model enables calculating the inlet flow of natural gas to the reformer needed for a certain plant size.

In modeling it is assumed that the overall voltage is the potential Gibb's electromotorical force minus the sum of the individual stack losses:

$$V = V_0 - \eta_{ohmic} - \eta_{act} \tag{16}$$

V_0 is the potential at zero current minus the concentration overpotential [V] governed by the Nernst-equation. The ohmic overpotential η_{ohmic} [V] is modeled as a linear function accounting for the various ohmic losses in the fuel cell membranes, gas diffusion layers and contact resistance.

The activation overpotential can be separated into the activation at the anode side and the cathode side. The anode activation overpotential is several orders of magnitude smaller that the cathode overpotential so it can be neglected. This yields the following coupled differential equations for calculating the cathode activation overpotential using Butler-Volmer kinetics and a homogeneous catalyst layer model. The cathode model uses the concept originally described by Springer et al., 1991:

$$\frac{dI}{dy} = t_{cat} \cdot i_{0,cathode} \left(\frac{c_{O_2}}{c_{0,O_2}} \right)^{\gamma_{cathode}} \cdot \left(\exp\left(\frac{\alpha_a F \eta_c}{RT} \right) - \exp\left(\frac{-\alpha_c F \eta_c}{RT} \right) \right) \tag{17}$$

$$\frac{d\eta_c}{dy} = \frac{I \cdot t_{cat}}{\sigma} \tag{18}$$

$$\frac{dC_{O_2}}{dy} = \frac{t_{cat} \cdot \left(I - I_{y=1} \right)}{n \cdot F \cdot D_{eff,O_2}} \tag{19}$$

The equations were non-dimensionalized multiplying with the catalyst thickness t_{cat} [m] using the catalyst layer coordinate y∈[0;1]. I is the current density [A/m²], C_{O2} is the oxygen concentration at the catalyst layer interface [mol/m^3], F is Faradays constant, n is the number of electrons involved in the reaction (n=4) and I.

Equations 17-19 is solved with initial values for the three constitutive variables and the boundary condition that the current density should be $I=I_{y=1}$ at y=1 (i.e. the Nafion surface after the catalyst layer). The initial cathode overpotential is fitted using a shooting method based on a combination of parabolic search and golden section search.

Integration is stopped when the oxygen concentration reaches zero in the catalyst layer. A linear correlation was used to estimate the overpotential at y=1, see fig. 8, which shows the variation of the constituents in the catalyst layer at I=0.8A/cm² – note that (19) has been non-dimensionalized prior to integration putting the initial oxygen concentration at the catalyst surface in the denominator.

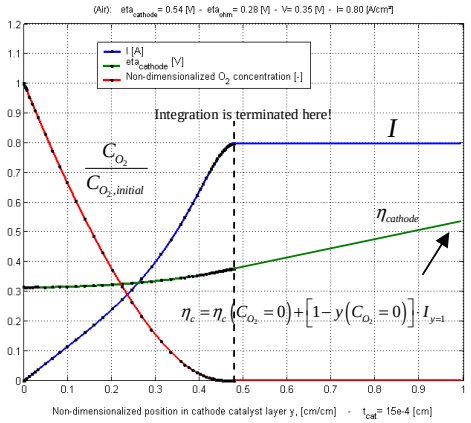


Fig. 8. Properties in catalyst layer (I=0.8A/cm²).

Several equations are needed to calculate the oxygen concentration at the catalyst layer surface and the effective diffusion coefficient $D_{eff,O2}$ in the catalyst layer material. A thorough description of the catalyst layer modeling is given by Marr & Li, 2000 in their paper "Composition and performance modeling of catalyst layer in a proton exchange membrane fuel cell". The additional parameters in the electrochemical model such as σ (proton conductivity [S/m]), i_0 (exchange current density [A/m²]) and α_a & α_c (anodic/cathodic transfer coefficient) can be found here as well. The exchange current density is a measure for the electrochemical activity of the catalyst layer, and it represents the current density for system equilibrium.

OPTIMIZATION OF HEN

Using the system model kinetics it is possible to establish the flow capacities needed (mass flow times specific heat capacity) for each flow. Temperature intervals are given by catalyst properties.
In order to optimize the heat exchanger network (HEN) the method of Yee and Grossmann [17] can

be used. The methodology is to minimize the total cost of the HEN, which can be described by the following overall objective function [17]:

$$\min(Total\ Cost) = Area\ Cost + Fixed\ Cost\ Units + Utility\ Cost \quad (20)$$

This method uses a simplified superstructure to find the optimum HEN-configuration allowing for stream splits. However, it is assumed that the mixing is isothermal. This means that the method is correct when no stream splits occur but if this is not the case, the method in some cases fails to find the optimum solution [18]. An approximation to the LMTD method is used to prevent numerical difficulties when equal approach temperatures on both sides of a heat exchanger occur. Yee and Grossmann proposed the Chen approximation (21).

$$LMTD = \frac{\Delta T_1 - \Delta T_2}{\ln\left(\frac{\Delta T_1}{\Delta T_2}\right)} \approx \sqrt[3]{\Delta T_1 \Delta T_2 \left(\frac{\Delta T_1 + \Delta T_2}{2}\right)} \quad (21)$$

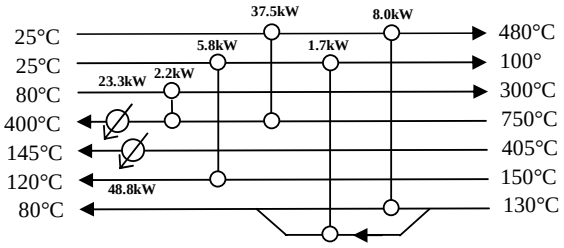
A simplified heat transfer model neglecting heat conduction was used. The objective function, which has to be minimized subject to appropriate constraints, was formulated by Yee and Grossmann as follows:

$$\min \sum_{i\in HP} CCU\ qcu_i + \sum_{j\in CP} CHU\ qhu_j + \sum_{i\in HP}\sum_{j\in CP}\sum_{k\in ST} CF_{ij} z_{ijk} + \sum_{i\in HP} CF_{i,CU} zcu_i +$$
$$\sum_{j\in CP} CF_{j,HU} zhu_j + \sum_{i\in HP}\sum_{j\in CP}\sum_{k\in ST} C_{ij}\left[\frac{q_{ijk}}{U_{ij}\sqrt[3]{\Delta T_{ijk}\Delta T_{ijk+1}\left[\frac{\Delta T_{ijk}+\Delta T_{ijk+1}}{2}\right]}}\right]^{\beta_{ij}} +$$
$$\sum_{i\in HP} C_{i,CU}\left[\frac{q_{qcu_i}}{U_{i,CU}\Delta T_{CU,i}\left(TOUT_i - TIN_{CU}\right)\sqrt[3]{\Delta T_{CU,i}\left[\frac{TOUT_i + TIN_{CU}}{2}\right]}}\right]^{\beta_{i,CU}}$$
$$\sum_{j\in CP} C_{HU,j}\left[\frac{q_{qhu_j}}{U_{HU,j}\Delta T_{HU,j}\left(TIN_{HU} - TOUT_j\right)\sqrt[3]{\Delta T_{HU,j}\left[\frac{TIN_{HU} + TOUT_j}{2}\right]}}\right]^{\beta_{j,HU}} \quad (22)$$

Feasibility of temperature interval matches and approach temperatures are applied as constraints to (22). The minimum approach temperature (pinch point) can be defined using a constraint as well.
The minimization of the MINLP problem produced by (22) can be performed using GAMS using for instance the MINLP solver DICOPT. This is an algorithm based on outer approximation for solving mixed-integer nonlinear programming (MINLP) problems that involve linear binary variables as well as linear and nonlinear continuous variables developed by professor developed by J. Viswanathan and Ignacio E. Grossmann, 1990.

The area cost of the heat exchangers was assumed to follow the simplified cost function:
Exchanger Cost [$] = Unit Price + C · (Area)$^{\beta}$

An example retrofit HEN using the temperature intervals on fig. 4 & heat capacities and flows corresponding to 100kWnet output is shown below:



It is seen that the need of hot utility has been eliminated from the system and cooling is reduced.

DISCUSSION

A model of a natural gas reforming fuel cell system has been established. Using the parameters found in modeling an optimal heat exchanger configuration can be found subject to the overall cost of the system. The method needs further investigation regarding stream splits and the afterburner subsystem should be included.

Moreover, the dynamics of the resulting HEN-networks should be investigated using dynamical models of the HEN and the reactors.

It is however expected that the present model will be a useful tool in system design in determining the optimum amount of heat integration for a given heat/power application. In near future it is expected to verify this experimentally at Aalborg University.

In the preliminary calculations, the Shift- and PrOx reactors were considered adiabatic. Heating these reactors externally should be investigated.

REFERENCES

[1] Rostrup-Nielsen J.R. *Catalysis Science and Technoogy* (Volume 5). Edited by Anderson J.R. and Boudart M. Springer-Verlag, 1984.

[2a] Xu J, Froment G.F. *Methane Steam Reform ing, Methanation and Water-Gas Shift: a. Intrinsic Kinetics*. AIChE J. 1989;35(88).

[2b] Xu J, Froment G.F. *Methane Steam Reforming, M. and WGS: b. Diffusional limi tations and Reactor Simulation*. AIChE J. 1989;35(97).

[3] Levent M., Budak G. *Estimation of concentration and temperature profiles for metane-steam reforming reaction in a porous catalyst*. Fuel Processing Tech. 1998;55.

[4] Rajesh J.K., Gupta, S.K., Rangaiah G.P., Ray A.K. *Multiobjective Optimization of Steam Reformer Performance Using Genetic Algorithm.* Ind. Eng. Chem. Res. 2000;39.

[5] Plehiers P.M., Froment G.F. *Coupled Simulation of Heat Transfer and Reaction in a Steam Reforming Furnace*. Chem. Eng. Tech. 1989;(12)20.

[6] Liese E.A., Gemmen R.S. *Transient Modeling Results Of A Methane Steam Pre refomer*. In AES-Vol 40, Proceedings of the SME Advanced Energy Systems Div. 2000.

[7] Froment G.F. *Production of synthesis gas by steam- and CO_2-reforming of natural gas.* Journal of Molecular Catalysis A 2000;163.

[8] Amphlett J.C., Mann R.F., Peppley N.A. *Onboard hydrogen purification for steam reforming PEM fuel cell vehicle power plants.* Int. J. of Hydrogen Energy 1995;21(8).

[9] *Fuel Cell Handbook (5^{th} Edition)*. By EG&G Services Parsons, Inc. Science Applications International Corporation. US Department Of Energy Department Of Fossil Energy 2000.

[10] Qi Z., He C., Kaufman A. *Effect of CO in the anode fuel on the performance of PEM fuel cell cathode.* Journal Of Power Sources 2002;4880.

[11] Froment G.F. Bischoff K.B. *Chemical Reactor Analysis and Design (2^{nd}. Edition)*. John Wiley & Sons Inc., 1990.

[12] Elnashaie S.S.E.H., Elshishini S.S. *Modelling, Simulation and Optimization of Industrial Fixed Bed Catalytic Reactors.* Top ics in chemical engineering Volume 7. Amersterdam: Gordoon and Breach Science Publishers, 1993.

[13] Mills A.F. *Heat And Mass Transfer.* Richard D. Irvin Inc., 1995

[14] Carey, G.F., Finlayson, B.A. *Orthogonal Collocation on Finite Elements.* Chemical Engineering Science 1975;(30) 587.

[15] Riitta L. Keiski, Tapio Salmi, Pekka Niemistö, Jorma Ainassaari and Veikko J. Pohjola. *Stationary and transient kinetics of the high temperature water-gas shift reaction.* Applied Ctalysis A: General, 1996;(137) 2.

[16] Yongtaek Choi and Harvey G. Stenger. Kinetics, simulation and insights for CO selective oxidation in fuel cell applications. Journal of Power Sources 2004;(129) 2.

[17] Lorentz T. Biegler, Ignacio E. Grossmann and Arthur W. Westerberg. *Systematic Methods of Chemical Process Design.* Prentice Hall, 1997.

[18] Floudas, C.A., *Nonlinear and Mixed-Integer Optimization: Fundamentals and Applications* Oxford University Press, New York, 1995.

[19] Julien Godat and Francois Marechal. *Optimization of a fuel cell system using process integration techniques.* Journal of Power Sources 2003;(118) 1-2.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

CALCULATION OF HEAT FLOW FROM BURIED PIPES USING A TIME-DEPENDENT FINITE ELEMENT MODEL

Tommy Persson* and Janusz Wollerstrand
Lund Institute of Technology
Department of Heat and Power Engineering
P.O. Box 118, SE-221 00 Lund
Sweden

Abstract

The paper describes a time-dependent model for calculating heat flow from buried pipes. The model is based on the finite element method. In order to minimize the influence of the fact that only a finite part of the ground is modelled, the ground block has to be chosen sufficiently large. The factors that mainly influence the choice of ground block size are: Media temperatures, ground composition and whether the pipes are insulated or not. For district heating pipes a ground block of 10 x 20 m is normally sufficient while corresponding dimensions for pipes used for heat gathering for heat pumps are 18 x 36 m. The convective heat transfer coefficient at the ground surface has a clear influence on the calculated heat flows. Considering errors introduced by the assumptions made, a default mesh created in FEMLAB® for the studied applications was found to be more than adequate. Although hard to investigate, ground soil properties are surely of significant importance to the calculated heat flows. This is especially the case if the medium temperature is close to the ground temperature. In the paper it is shown that, particularly when studying cases with media temperatures close to the ground temperature, stationary models can produce results that differ greatly from time-dependent calculations.
Keywords: Finite element method, heat flows, buried pipes, time-dependent model

Nomenclature

Abbreviations

a	Thermal diffusivity [m^2/s]
b, c, d	Distances [m]
c	Specifik heat [kJ/kgK]
DH	District heating
f	A general function
$\dot{Q}$	Heat flow [W]
T	Temperature [°C]
U	Overall heat transfer coefficient [W/K]
α	Convective heat transfer coeff. [W/m^2K]
ρ	Density [kg/m^3]
λ	Conductive heat transfer coeff. [W/mK]
σ	Average deviation between results [%]
τ	Time [days]

Subscripts

air	Property related to air
air/ground	Surface between air and ground
casing	Property related to casing pipe
ground	Property related to ground
ins	Property related to insulation
pipe	Property related to medium pipe
ret	Return medium (low temperature)
sup	Supply medium (high temperature)
water/pipe	Surface between water and pipe

Introduction

Heat flows to and from buried pipes are of interest in a number of applications. Two common fields are district heating and heat gathering for heat pumps. In a district heating system heat flows from the pipes are adverse to system performance. In the case with heat pumps, during wintertime heat extraction from the ground gives a heat contribution to the building and during summertime heat could potentially be rejected to the ground in order to obtain a cooling effect on the building. Thus, a low resistance to heat transfer is positive in this case. The aim of this paper is to provide the reader with information needed to

*Corresponding author. Phone: +46 46 22 29273,
Fax: +46 46 22 24717, Tommy.Persson@vok.lth.se

perform accurate heat flow calculations for buried pipes. The calculations are based on the Finite Element Method (FEM). The commercial software FEMLAB® was used for the calculations. For details on the theory of the FEM the authors refer to Ottosen et al [1].

Description of a time-dependent FEM-model

The geometry for the FEM-model and definitions used in the paper are shown in Figure 1.

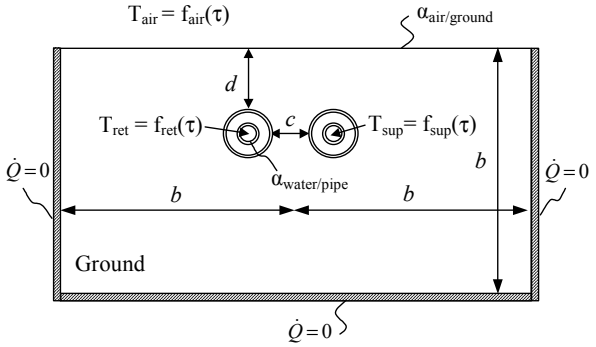


Figure 1 Geometry for FEM-model

The model consists of a ground block of size b x $2b$ metres and two pipes located in the horizontal centre of the block with an internal distance of c metres. The distance from the pipe head to the surface is d metres. In Figure 1 the pipes are shown as DH pipes and are thus insulated. The details for the DH pipes used in the calculations are shown in Figure 2.

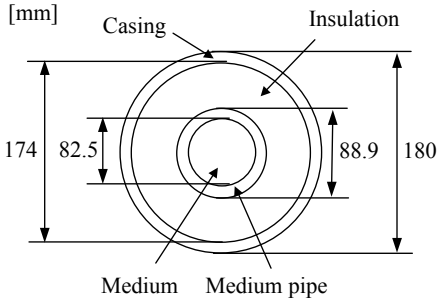


Figure 2 Details for DH pipe

The air and media temperatures are functions of time. The boundaries towards the air and the media are set as convective. Both the convective heat transfer coefficient at the ground surface and at the pipe walls are regarded to be constant. The boundaries between the ground block and the surrounding ground are set as perfectly insulated (i.e. symmetry). Unless otherwise is stated the assumptions according to Table 1 and Table 2 are used throughout the paper.

Property	Value
Convective heat transfer coefficient, air/ground, $\alpha_{air/ground}$	15 W/m^{2}K
Convective heat transfer coefficient, water/pipe, $\alpha_{water/pipe}$	2500 W/m^{2}K
Size of ground block, b	10 m
Distance between pipes, c (insulated pipes)	0.2 m
Distance between pipes, c (uninsulated pipes)	2 m
Pipe depth, d (insulated pipes)	0.6 m
Pipe depth, d (uninsulated pipes)	1.5 m

Table 1 Assigned values to various properties

Material	Heat conductivity, λ, [W/mK]	Density, ρ, [kg/m^3]	Heat capacity, c, [J/kgK]
Pipe (steel)	76	7850	480
Insulation (PUR)	0.030	60	1700
Casing (PEH)	0.43	940	1900
Ground (moraine)	1.5	1800	1200

Table 2 Thermal properties of pipe, insulation, casing and ground

Since the scope of this paper is to investigate the influence of various assumptions on the calculated heat flows it is easier to interpret the results if the material properties are constant, see Table 2. However, when performing calculations for a specific case dependence on temperature, time or location should be regarded and can easily be implemented in FEMLAB®.

In the model, only heat conduction in the ground is considered. Based on a discussion in Sundberg [2] it is concluded that conduction is the most important heat transfer mechanism in most types of soil. The largest error related to this simplification is probably due to the fact that energy bound/released at the phase-change ice/water is neglected.

Initialising the model

In order to obtain a realistic temperature distribution in the ground at the beginning of all calculations in the paper, an initial calculation covering a period of 4 years is performed. The

states at the end of this calculation are then mapped as the initial state to the calculations that are analysed. The FEMLAB® commando "*asseminit*" is used for the mapping.

Ground temperature distribution in reality and in model

A description of stationary calculations of heat losses from DH pipes is found in Jonson [3]. The results presented there are very close to specifications from pre-fabricated district heating pipe manufacturers. The main objection to this kind of calculations is that the problem, which is clearly time-dependent, is treated as a stationary one. As a consequence of the time-dependent reality a correct ground temperature distribution can never be obtained using a stationary model.

Real ground temperature distribution

Figure 3 shows a typical yearly range of ground temperature variation at different depths. In this case the ground temperature varies around an average value of 8 °C. Due to the thermal inertia of the ground the temperature at the surface varies between -4 – +23 °C while at depths below 6 m it is relatively constant. The exact details of how the temperature will vary with depth during a year depends on the composition of the ground, as will be shown below, and the climate.

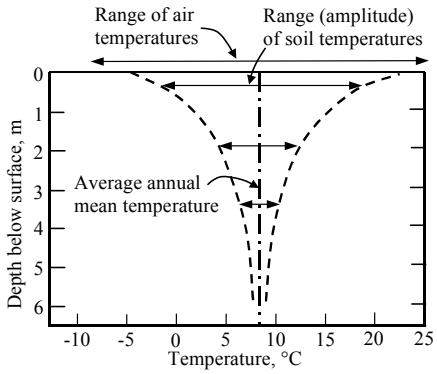


Figure 3 Ground temperature variations during a year in Ottawa, Canada, adapted from Williams & Gold [4]

Modelling the ground

If a material is considered as homogenous (i.e. $\lambda(x,y,z)$ = constant) and there is no internally generated heat, the equation of heat conduction can be simplified to:

$$\frac{\partial T}{\partial \tau} = \frac{\lambda}{\rho c}\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2}\right) = \frac{\lambda}{\rho c}\nabla^2 T = a\nabla^2 T \quad (1)$$

The term ($\lambda/\rho c$) is called the thermal diffusivity and is denoted a. According to (1) the thermal diffusivity is a measure of how the temperature in a material changes with time, τ, when exposed to a disturbance. In Figure 4 is shown how the temperature varies with depth according to FEM-calculations in two extreme cases approximately representing dry moraine and granite with a high fraction of quartz, respectively, as specified in Table 3. The model shown in Figure 1, apart from the pipes, with b=20 m was used.

Material	λ, [W/mK]	ρ, [kg/m^3]	c, [J/kgK]
Moraine, dry	0.6	1800	1600
Granite, high fraction of quartz	4	2700	800

Table 3 Thermal properties of ground, extreme cases

The air temperature in Figure 4 was varied as:

$$T_{air} = 7 + 11\sin\left(\frac{2\pi\tau}{365}\right) \quad (2)$$

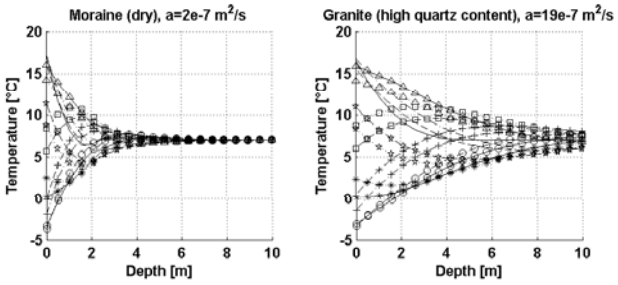


Figure 4 Calculated ground temperature variations during a year

Figure 4 shows that the thermal properties of the ground significantly influences how the air temperature penetrates the ground. For instance, in the left case, the ground temperature is essentially constant at a depth of 6 m while for the right case it varies with an amplitude of about 2.5 °C.

An alternative way to display the results from the calculations is to study how the temperature varies with time at different depths. In Figure 5, temperature variations at depths 0-10 m is shown.

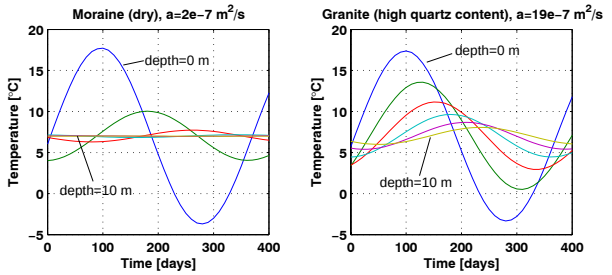


Figure 5 Calculated ground temperature variations at depths of 0, 2, 4, 6, 8 and 10 m

In Rosén et al [5] it is stated that temperature variations in the ground normally decrease exponentially with depth and that it can be compared to harmonic oscillations that are dampened and phase shifted compared to the temperature variations at the ground surface. This is a good description of the results in Figure 5 and thus it seems that the model produces realistic results.

Influence of introduced assumptions

In this section the influence of some of the assumptions made will be investigated. All calculations are based on the model in Figure 1. Since the influence of the assumptions made may depend on the situation studied three different cases, according to Table 4, will be regarded. The cases were chosen in order to compare some qualitatively different situations. Considering real applications the first two cases can be taken to approximate high/low temperature DH while the third case corresponds to heat gathering from the ground for a heat pump.

Case nbr	Insu-lation	Supply temperature	Return temperature
1	Yes	$110-10\sin\left(\frac{2\pi\tau}{365}\right)$	$40+5\sin\left(\frac{2\pi\tau}{365}\right)$
2	Yes	$70-10\sin\left(\frac{2\pi\tau}{365}\right)$	$20+5\sin\left(\frac{2\pi\tau}{365}\right)$
3	No	$2+5\sin\left(\frac{2\pi\tau}{365}\right)$	$6\sin\left(\frac{2\pi\tau}{365}\right)$

Table 4 Description of studied cases

The studied period is one year and the heat flows are calculated every 48 hours. The deviation in calculated results between a case k and a reference case is estimated as:

$$\sigma = \frac{1}{N \cdot \left| mean\left(\dot{Q}_{ref,i}\right)\right|} \sum_{i=1}^{N} \left| \dot{Q}_{k,i} - \dot{Q}_{ref,i} \right| \qquad (3)$$

According to (3) σ represents the average deviation between the reference case and case k in percent of the mean value in the reference case. Thus, as σ approaches zero the difference between case k and the reference case is small.

The calculated net heat flows from the return and supply pipes for the three cases in Table 4 are shown in Figure 6.

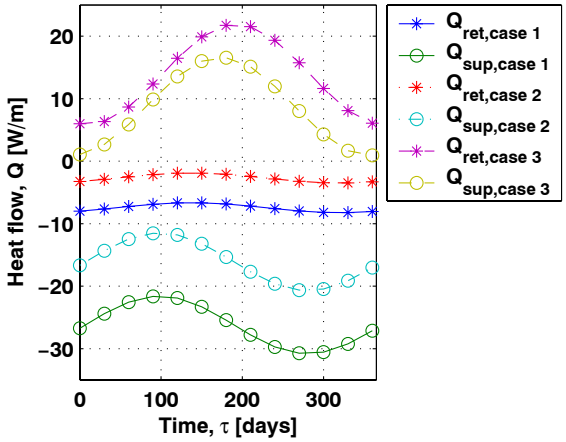


Figure 6 Net heat flows from pipes in three cases

In Figure 6 a net heat flow from a pipe is assigned a negative value while a net heat flow to a pipe is positive. For the cases with insulated pipes (cases 1 and 2) there is a net heat flow from the pipes to the surrounding. The heat losses are larger in case 1 than in case 2 due to the higher temperatures in the first case. Since the distance between the pipes is only 0.2 m the temperature fields around the two pipes coincides. This causes the heat losses to be smaller compared to if the two pipes are placed at a large distance from each other. In the case with the uninsulated pipes, case 3, the net heat flow is positive for both the return pipe and the supply pipe. Since the temperature in the supply pipe is closer to the surrounding ground temperature the net heat flow to this pipe is lower than to the return pipe.

Ground block size

In order to reduce the influence of the finite ground block studied it has to be chosen sufficiently large. On the other hand a larger ground block will increase the computational time and thus it should not be unnecessarily large. In Figure 7 it is shown how the calculated heat loss in the three cases changes as the distance b in Figure 1 is varied. For all cases the calculated heat flows at b = 20 m is used as reference in (3).

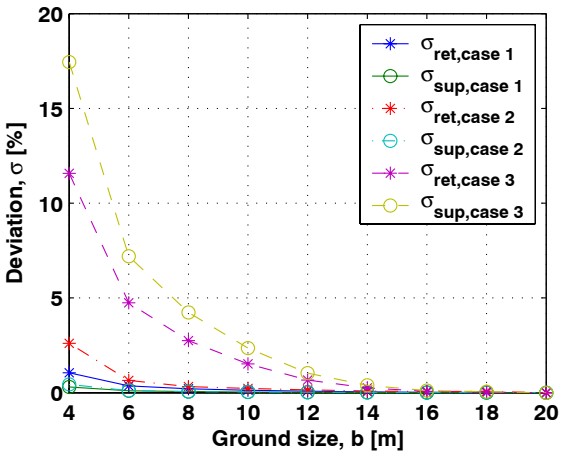


Figure 7 Change in calculated heat flows as a function of *b*

The overall tendency of the curves in Figure 7 is that the derivative of $\sigma(b)$ decreases as *b* increases. I.e. the incremental gain, considering accuracy, with increasing the ground size block becomes smaller for large values of *b*. Considering the gentle slope of the curves to the right of Figure 7 the calculated heat flows at $b = 20$ m must be close to the value when *b* is infinity and is therefore considered as the 'true' values. When the results in Figure 7 are studied it should be remembered that the distance between the pipes is 2 m in case 3 and 0.2 m in cases 1 and 2.

According to Figure 7 the difference between the cases is large regarding the influence of the ground block size, *b*. For instance, the deviation σ is below 0.7 % at $b > 6$ m for cases 1 and 2, while for case 3 *b* has to be chosen as > 14 m to obtain this. The large difference in required *b* value cannot be explained solely by the difference in internal pipe distance. First, in cases 1 and 2 the pipes are insulated, and since the temperature gradient is steep within the insulation the influence of the pipes on the surrounding ground temperature is small. Second, since the temperature difference between the media in the uninsulated pipes and the surrounding ground is small, an incorrect ground temperature will have a large influence (in percent) on the calculated heat flows. The magnitudes of the temperature difference is also the reason that σ is greater for the return pipe than for the supply pipe in cases 1 and 2 and the opposite in case 3. Particularly in cases 1 and 2 the greater deviation for the return pipes can also be explained by the fact that heat flows from the supply pipe to the return pipe. As the ground block around the pipes is increased the influence of the insulated boundaries decreases, and therefore also the heat flow between the media decreases.

Finally, considering the great influence the ground properties has on the ground temperature distribution, according to Figure 4, the selection of the distance *b* must be influenced by the ground properties. This will be studied later on in this paper.

Mesh size

The fundamental idea of the finite element method is to divide the geometry into small elements. In each element a rather crude estimation of how the property varies within the element is made. If the size of the elements is too large this can introduce errors. For this reason it is of interest to investigate how the calculation results changes when the element size is decreased. The default mesh generated by FEMLAB® consists of 18568 elements in cases 1 and 2 and 6650 elements in case 3 (due to the less complex geometry). The refined meshes consist of 74272 and 26600 elements, respectively. Due to lack of data storage capacity the computer used (having 736 MB RAM) could not compute cases 1 and 2 using the refined mesh and therefore only case 3 could be investigated.

As the mesh was refined in case 3 the deviation, σ, between calculated heat flows using a refined mesh and the default mesh was 0.091 % for the return pipe and 0.061 % for the supply pipe. Considering the influence of other assumptions made and the increase in computational time when using a refined mesh the default mesh created in FEMLAB® is more than adequate. It is assumed that the general result also holds for cases 1 and 2.

Boundary conditions

Regarding the choice of boundary conditions the convective boundaries at the ground surface and inside the media pipes is a natural choice. Figure 8 and 9 demonstrate how the calculated heat flows are influenced by the values of $\alpha_{water/pipe}$ and $\alpha_{air/ground}$. The heat flows at $\alpha_{water/pipe} = 4500$ W/m^{2}K and $\alpha_{air/ground} = 25$ W/m^{2}K are used as reference values in (3).

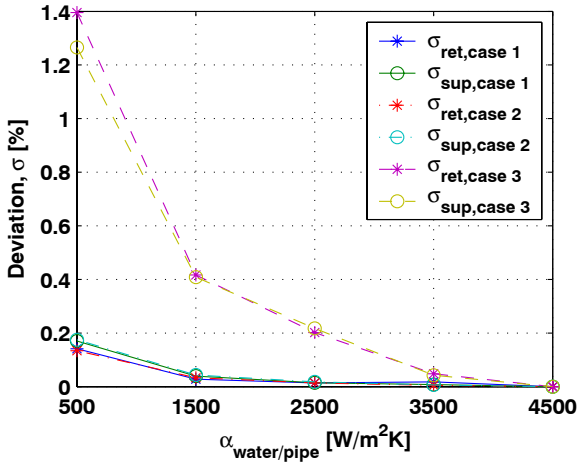


Figure 8 Change in calculated heat flows as a function of $\alpha_{water/pipe}$

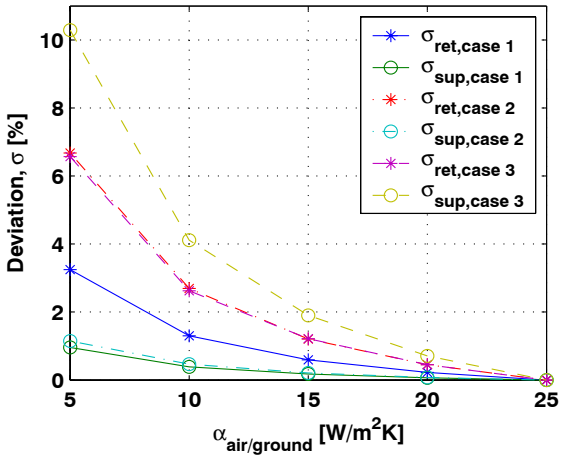


Figure 9 Change in calculated heat flows as a function of $\alpha_{air/ground}$

If a single pipe in the ground is considered the heat flow from it can, simplified, be expressed as:

$$\dot{Q} = U_{tot} \cdot \Delta T_{media/surr} = \frac{\Delta T_{media/surr}}{\frac{1}{U_{\alpha_{water/pipe}}} + \frac{1}{U_{pipe}} + \frac{1}{U_{ins}} + \frac{1}{U_{casing}} + \frac{1}{U_{ground}} + \frac{1}{U_{\alpha_{air/ground}}}} \quad (4)$$

For a well-insulated pipe, $1/U_{ins}$ is dominating in the denominator in (4) and therefore a change in the heat resistance for the other components will only have a limited influence on the heat flow. For this reason the influence of $\alpha_{water/pipe}$ is negligible in cases 1 and 2. For a pipe with no insulation the term corresponding to insulation in (4) is not present and therefore the influence of $\alpha_{water/pipe}$ is larger (but still small). As $\alpha_{water/pipe}$ and $\alpha_{air/ground}$ is increased it holds that the incremental influence they have on U_{tot} decrease. For this reason the derivative of the curves in Figure 8 and Figure 9 decrease when moving to the right.

The value of $\alpha_{air/ground}$ has a much greater influence on the calculated heat flows than $\alpha_{water/pipe}$. This is partly due to the fact that $\alpha_{air/ground}$, according to (4), influences U_{tot}, but apart from this it also indirectly influences the ground temperature. The closer the media temperatures are to the ground temperature the greater the influence of $\alpha_{air/ground}$, i.e. the ground temperature, will be on the calculated heat flows. This trend is apparent when studying the results in Figure 9.

At large distance (in all directions) from the pipes the ground temperature distribution is not influenced by the presence of the pipes. If a cut is made in the undisturbed ground, the temperature is exactly the same on both sides of the cut. In other words, symmetry prevails. Thus, considering that perfectly insulated boundaries also can be regarded as symmetry planes the vertical ground boundaries in Figure 1 should be set as $\dot{Q} = 0$. The boundary condition on the lower ground boundary can be chosen either as $\dot{Q} = 0$ or as a constant temperature (equal to the average temperature at the depth). As long as the ground block is chosen to be of sufficient depth, these alternatives will produce the same result. An advantage with using a perfectly insulated lower boundary is that it is not necessary to know the average temperature of the soil. Apart from this it is also easier, by studying if the temperature variations at the bottom of the ground block is relatively constant or not (compare with Figure 4), to determine if the ground depth is set sufficiently large when using $\dot{Q} = 0$.

Ground properties

Considering the large influence the properties of the ground has on the ground temperature distribution, according Figure 4, the calculated heat flows must also depend on ground material properties. In Table 5 the calculated heat flows in cases with ground properties according to Table 2 (reference cases) is compared to calculated heat flows using the quite extreme ground properties in Table 3.

Case	σ_{return} [%]	σ_{supply} [%]
1, moraine	24.0	11.0
1, granite	13.3	5.7
2, moraine	33.9	10.5
2, granite	19.0	5.4
3, moraine	58.5	57.8
3, granite	155.5	151.3

Table 5 Deviation between calculated heat flows depending on ground properties

According to the results in Table 5 the ground properties have a large influence on the calculated heat flows. As the thermal diffusivity is increased the temperature variations at a given depth will increase, Figure 4. Thus the heat flow to/from the pipes will increase during some periods and decrease during others. Apart from this the insulating capacity for granite is lower than for moraine and this also affects the resulting heat flows. From Table 5 it can be concluded that the assumed ground properties for most applications is of great importance to the calculation results. Unfortunately the exact ground composition is often time consuming to investigate.

As the influence of the ground properties is studied we will return to the previous discussion regarding the selection of the ground block size. A temperature source will influence a greater part of the ground as the thermal diffusivity of the ground is increased. Considering this it is relevant to study how the results in Figure 7 are affected if the reference ground is replaced by granite with a high fraction of quartz (Table 3), see Figure 10.

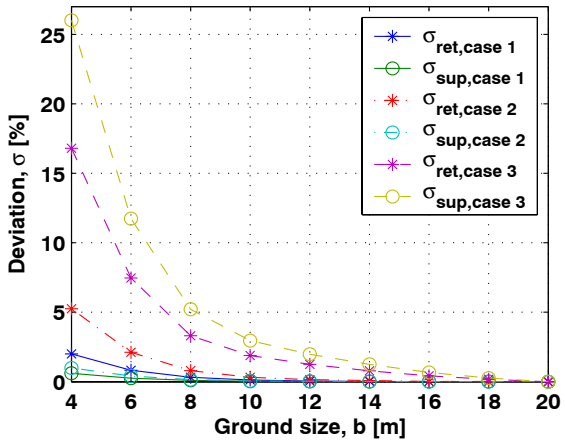


Figure 10 Change in calculated heat flows as a function of *b*, ground with large thermal diffusivity

Stipulating that the deviation, σ, should be below 0.5 % for both the supply and return pipe, the required distances *b* according to Figure 7 is 6, 8 and 14 metres in cases 1, 2 and 3, respectively. The corresponding *b* values in Figure 10 are 8, 10 and 18 metres. Thus, as the thermal diffusivity of the ground is increased the ground block should be chosen larger. However, it should be pointed out that the assumed thermal diffusivity of the granite in Figure 10 is very high and considering this a distance *b* in Figure 1 of 16-18 metres should be sufficient for most cases.

Comparison between stationary and time-dependent models

In this section the time-dependent model is compared to a stationary model. Heat loss calculations are performed using the air and media temperatures in (2) and Table 4. In the stationary model heat flows are calculated using momentary values of the temperatures. In the time-dependent model the heat flows are calculated continuously. The deviation in calculated heat flows between the stationary and the time-dependent model, using the time-dependent model as reference, is shown in Table 6.

Case nbr	σ_{return} [%]	σ_{supply} [%]
1	8.2	2.3
2	22.5	3.7
3	59.9	93.7

Table 6 Comparison of calculated heat flows using a time-dependent or a stationary model

From the results in Table 6 it is obvious that a stationary model can introduce significant errors. The deviation in percentage is greater the closer to the ground temperature the media temperature is. For this reason the deviation in calculated heat flows between the stationary and the time-dependent model is very large in case 3 and much less in case 1. While the assumption of stationary conditions in some situations, at best, is acceptable, case 1, it might be catastrophic in other applications, case 3.

Using measured data as input to FEM-model

A nice feature in FEMLAB® is that it is possible to use measured data as input to the FEM-model and

thereby simulating real situations. An example of this can be found in Persson & Wollerstrand [6] where heat losses from a DH culvert were calculated using measured media and air temperatures.

When implemented in FEMLAB®, both the value of the measured property and the derivative of the property must be defined. Both values can be specified using a look-up table and the Matlab commando *"flinterp1"*. When using look-up tables it is important to choose a sufficiently short sample time. In our case the "measured" properties varies as sinus waves with a period of 365 days. In Figure 11 the calculated heat flows using look-up tables with different samples times are compared to the corresponding values when the analytical expressions in (2) and Table 4 are directly implemented in FEMLAB®.

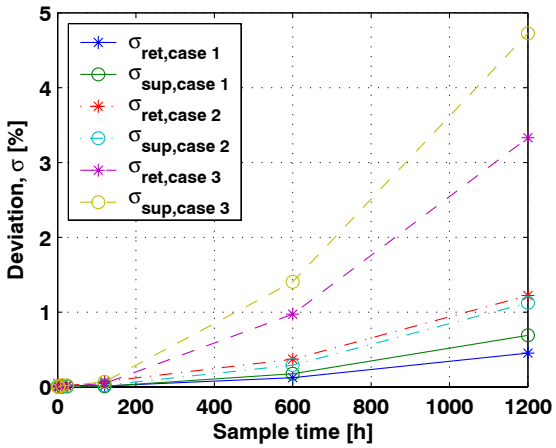


Figure 11 Comparison of calculated heat flows using analytical expressions and look-up tables

According to Figure 11, the sensitivity to sample time is dependent on the situation studied. A general trend seems to be that when the amplitude of the variations is large, compared to the mean value of the property, a short sample time should be chosen. Considering computational times, see below, it is advised to use as short sample times as possible and at least not greater than 1/30 of the period of the variations of the measured properties.

Computational times

Finally, some comments will be made on the computational times for the three cases studied using different models. The computational times are displayed in Table 7. The computations were performed on an Intel Pentium 4 2.4 GHz with 736 MB RAM and Windows XP as operating system. Version 2.3 of FEMLAB® was used.

Type of model	Case 1	Case 2	Case 3
Reference case	2024	2024	750
b=20	2160	2090	801
Refined mesh	—	—	3615
Stationary model	2171	2163	876
Look-up table, sample time 12 h	2084	2082	773
Look-up table, sample time 1200 h	6673	6295	2675

Table 7 Computational times in seconds for different cases using different models

According to Table 7 the computational time is not affected by the temperature difference in cases 1 and 2. Neither has the size of the ground block a significant influence on the computational time. The less complex geometry of case 3 is significantly faster to compute than cases 1 and 2. Considering this, and the certainly small influence of the pipe and the outer casing, the DH pipes in Figure 2 could be reduced to only consisting of the insulation if computational time is crucial. Using look-up tables with a relatively short sample time only has a small effect on the computational time while greater sample times actually increases it. The authors' guess is that this is due to the occasionally rapid changes of the derivative when stepping through the look-up table.

Using a refined mesh greatly increases the computational time. The computational time using a stationary model is actually almost identical to the time-dependent model. However, since the time-dependent model always has to calculate for the whole period studied as well as for an initialising period the comparison is greatly dependent on the sample time. The computational time decreases when using the stationary model if the sample time is increased.

Conclusions

Factors that influence the choice of ground block size are: Media temperatures, ground composition and whether the pipes are insulated or not. For DH pipes a ground block of 10 x 20 m should be sufficient, while the corresponding value for cases with heat gathering for heat pumps from the ground is 18 x 36 m. The value of the convective heat transfer coefficient at the ground surface has a

clear influence on the calculated heat flows. Of the factors studied in this paper the ground properties has the greatest influence on the calculated heat flows. A stationary model may produce results that differ greatly from those calculated with a time-dependent model. This is especially the case if the media temperatures are close to the ground temperature. When using look-up tables in FEMLAB® the sample time should be chosen small considering both accuracy and computational time.

References

[1] Ottosen, N., Petersson, H., *Introduction to the finite element method*, University of Lund, Sweden, Prentice Hall, ISBN 0-13-473877-2

[2] Sundberg, J., *Thermal properties of soils and rocks*, PhD thesis, Department of geology, Chalmers University of Technology, Gothenburg, Sweden, 1988, ISSN 0348-2367, Publ. A 57

[3] Jonson, E., *Värmeförluster från fjärrvärmenät i småhusområden -Inverkan av rörgeometri och dimensioneringskriterier [Heat losses from district heating systems in detached house areas -Influence of pipe geometry and design criteria],* Licentiate thesis, Department of Heat and Power Engineering, Lund University, Sweden 2001

[4] Williams, G.P., Gold L.W., *CBD-180 Ground Temperatures*, http://irc.nrc-cnrc.gc.ca/cbd/cbd18–0e.html, 2003-12-10

[5] Rosén, B., Gabrielsson, A., Fallsvik, J., Hellström, G., Nilsson, G., *System för värme och kyla ur mark -En nulägesbeskrivning [Ground source heating and cooling systems - A state of the art report]*, Swedish Geotechnical Institute, Linköping, 2001, ISBN 1100-6692, ISRN SGI-VARIA--01/511--SE

[6] Persson, T., Wollerstrand, J., *Heat losses from a novel type of district heating culvert*. In: Building Energy Systems, Proceedings of 17th International Conference on Efficiency, Costs, Optimisation, Simulation and Environmental Impact of Energy and Process Systems, Mexico, 2004

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

DYNAMIC MODELING OF A DOMESTIC HOT WATER SYSTEM USING SIMULINK®

Tommy Persson* and Janusz Wollerstrand
Lund Institute of Technology
Department of Heat and Power Engineering
P.O. Box 118, SE-221 00 Lund
Sweden

Abstract

The paper describes a dynamic model of a domestic hot water circulation system. The components of the system are described in mathematical terms and it is shown how they are implemented in Simulink®. The components studied are: pipes, self-acting thermostatic valves, circulation pumps and heat exchangers. Apart from this, two methods to calculate flow distribution in systems with extraction points are described. The first method is a further development of an explicit method based on Kirchoff's law for calculating flow distributions in closed systems, Persson [1]. The second method is based on the fact that the net pressure drop in a closed loop is zero (known as the Hardy-Cross method). To reduce computational time both methods should involve two flow calculation modes, one iterative and one explicit, depending on if flow is extracted from the system at the current time step or not. Flow calculations are established primarily in a form adapted for implemention in Simulink®. When numerical problems arise an equation solver in Matlab® can be used since this code, in contrast to the iterative solver within Simulink®, can be supplied with an initial guess, thus making the solver more stable. Considering computational time the Matlab® call-back should be converted to an S-function. It is also shown that the application Real-Time Workshop® greatly speeds up the calculations.
Keywords: Domestic hot water circulation, Simulink, dynamic model, flow calculation

Nomenclature

Abbreviations

A	Area [m^2]
c_p	Specific heat [kJ/kgK]
DHW	Domestic hot water
e	Control error [°C]
F	Ratio [-]
k_v	Flow capacity [m^3/h]/$bar^{0.5}$
k_{vs}	Maximum flow capacity at $\Delta p = 1$ bar [m^3/h]
m	Mass [kg]
$\dot{m}$	Mass flow [kg/s]
N	Number of branches [-]
Δp	Pressure difference [Pa]
T	Temperature [°C]
t	Time [s]
U	Overall heat transfer coeff. [W/mK]
V	Volume [m^3]
$\dot{V}$	Volume flow [m^3/s]
X_{prop}	Proportional band [°C]
z	Valve opening degree [%]
α	Convective heat transfer coeff. [W/m^2K]
v	Velocity [m/s]

Subscripts

br	Branch
c	Cold side
eq	Equivalent
h	Hot side
i	Inside
in	Ingoing property
o	Outside
out	Outgoing property
p	Pipe
ret	Return
sup	Supply
$surr$	Surrounding
T	Temperature
$transp$	Transport
w	Water
$wall$	Heat exchanger wall

*Corresponding author. Phone: +46 46 22 29273,
Fax: +46 46 22 24717, Tommy.Persson@vok.lth.se

Introduction

A typical domestic hot water (DHW) system in a residential building consists of a DHW heater (1), distribution pipes (2), circulation pipes (3), a circulation pump (4) and connecting pipes (5), Figure 1. The water is circulated through a number of risers situated at different distance from the heater in the building. As pressure losses increase with increasing distance from the heater, the water flow distribution among the branches will be uneven unless hydraulic balancing has been employed. If the same temperature level on the return from all the branches is to be kept, the circulating flow in the outermost branches must be higher than the flow in the branches closer to the heater. This is due to the temperature drop along the system caused by heat losses. This topic is treated more extensively in Wollerstrand & Persson [2].

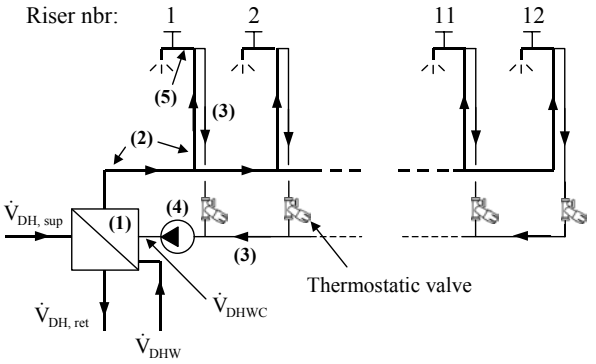


Figure 1 Principal scheme of a DHW circulation system

Description of the components in a DHW circulation system

The DHW system studied in this paper is shown in Figure 1: It consists of a heat exchanger, pipes, thermostatic valves and a pump. In this section these components are described mathematically and it is shown how they are implemented in Simulink®. The mathematical descriptions are based on previous work by the authors and by others as well; for a more detailed discussion cf. Persson [1], Gummérus [3] and Hjorthol [4].

Pipe model

Both when modeling pipes and heat exchangers the components are divided into a number of sections. By writing energy balances for each section the components can be modeled. First the water in a pipe section is studied. By formulating a heat balance for the water in a pipe section and then differentiating it the following expression is obtained, Persson [1]:

$$\frac{\partial}{\partial t}\left(T_{w,out}\right)=\frac{1}{m_w c_{p,w}}\left[\dot{m}_w c_{p,w}\left(T_{w,in}-T_{w,out}\right)-\alpha_{w\text{-}p}A_{i,p}\left(\frac{T_{w,in}+T_{w,out}}{2}-T_p\right)\right] \quad (1)$$

Next an energy balance for the pipe wall can be formulated, and by differentiating we obtain:

$$\frac{\partial}{\partial t}\left(T_p\right)=\frac{1}{m_p c_{p,p}}\left[\alpha_{w\text{-}p}A_{i,p}\left(\frac{T_{w,in}+T_{w,out}}{2}-T_p\right)-\alpha_{p\text{-}surr}A_{o,p}\left(T_p-T_{surr}\right)\right] \quad (2)$$

In (2) heat conduction within the pipe wall is disregarded and the mean temperature difference between the medium and the pipe is based on the arithmetic mean value of the inlet and outlet water temperatures. By modeling the pipe according to (1) and (2) it is assumed that the water is being perfectly stirred. One negative aspect of this assumption is that water having the inlet temperature when entering the pipe will have an immediate influence on the outlet temperature. This is not physically correct and therefore the inlet temperature should be delayed. The transport time for the flow through the pipe can be calculated as, Persson [1]:

$$t_{transp}=\frac{V_p}{\dot{V}} \quad (3)$$

However, due to the heat exchange between the flowing water and the pipe wall the temperature front will propagate somewhat slower than t_{transp}. As a measure of the difference in transport velocity between the temperature front, v_T, and the flow, v, Larsson [5] defines:

$$F=\frac{v_T}{v} \quad (4)$$

F is <1 and decreases as the ratio between the pipe wall thickness and inner diameter increases. According to (4), the transport time for the temperature front is obtained by dividing the right hand side of (3) by F. In Simulink® this can be implemented as shown in Figure 2.

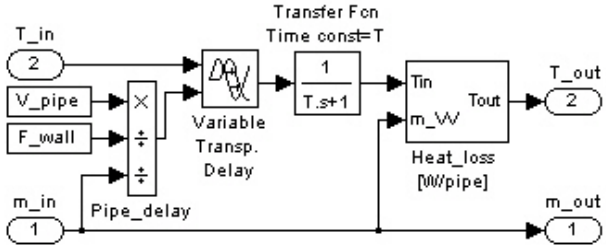


Figure 2 Implementation of transport time for temperature front in Simulink®

Treating the pipe wall as given by (1) and (2) results in a significant computational time. Thus, depending on the type of investigation made it could be desirable to make a simplification and disregard the pipe wall. Equations (1) and (2) are thereby replaced by:

$$\frac{\partial}{\partial t}\left(T_{w,out}\right)=\frac{1}{m_w c_{p,w}}\left[\dot{m}_w c_{p,w}\left(T_{w,in}-T_{w,out}\right)-U\left(\frac{T_{w,in}+T_{w,out}}{2}-T_{surr}\right)\right] \quad (5)$$

In Figure 3 is shown how (5) can be implemented in Simulink®. In (5) a Logarithmic Mean Temperature Difference, LMTD, between the medium and the surroundings can also be used. If the surrounding temperature, T_{surr}, is regarded as constant a look-up table can be used for deciding LMTD. The use of a look-up table is advantageous considering computational time.

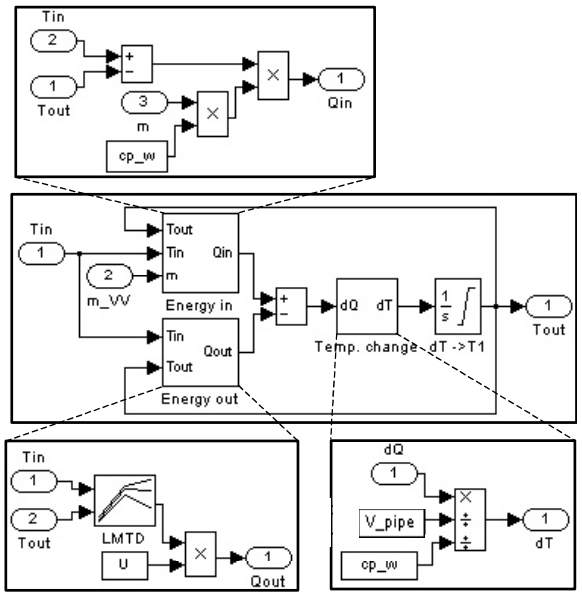


Figure 3 Implementation of (5) in Simulink®

Since the pipe wall is not considered in Figure 3 a *"Transfer Fcn"*-block, as shown in Figure 2, can be used for introducing some additional inertia to the system. The physical interpretation of this time constant is that it represents the thermal inertia of the pipe wall.

Heat exchanger model

The mathematical formulation of a heat exchanger and a pipe is quite similar. For the heat exchanger, equations must be formulated for hot water flow, cold water flow and for the wall separating them. By formulating energy balances for these three parts and differentiating the following equations are obtained:

$$\frac{\partial}{\partial t}\left(T_{c,out}\right)=\frac{1}{m_c c_{p,c}}\left[\dot{m}_c c_{p,c}\left(T_{c,in}-T_{c,out}\right)-\alpha_c A\left(\frac{T_{c,in}+T_{c,out}}{2}-T_{wall}\right)\right] \quad (6)$$

$$\frac{\partial}{\partial t}\left(T_{h,out}\right)=\frac{1}{m_h c_{p,h}}\left[\dot{m}_h c_{p,h}\left(T_{h,in}-T_{h,out}\right)-\alpha_h A\left(\frac{T_{h,in}+T_{h,out}}{2}-T_{wall}\right)\right] \quad (7)$$

$$\frac{\partial}{\partial t}\left(T_{wall}\right)=\frac{1}{m_{wall} c_{wall}}\left[\alpha_h A\left(\frac{T_{h,in}+T_{h,out}}{2}-T_{wall}\right)-\alpha_c A\left(\frac{T_{c,in}+T_{c,out}}{2}-T_{wall}\right)\right] \quad (8)$$

Considering the fact that many of the terms in (6) –(8) are identical, the equations can quite easily be implemented in Simulink®. If reduced computational time is considered essential mainly elementary blocks such as *"Sum"* and *"Product"* should be used. The temperature fronts for the hot and cold water flows within the heat exchanger should be delayed, as in the case with the pipe in Figure 2.

When approximating the temperature profile to be linear, as in (6) – (8), the heat exchanger must be divided into sufficiently many sections so that the theoretically correct temperature profile, which is logarithmic, can be accurately estimated. However, the computational time increases with the number of sections and for this reason not too many sections should be used. For most applications a suitable choice is to divide the heat exchanger into 3-5 sections, Persson [1].

Pump and thermostatic valve models

Thermostatic valves used in DHW circulation systems are normally self-acting and linear. In Figure 4 is shown how such a valve can be modeled. This model will be discussed next.

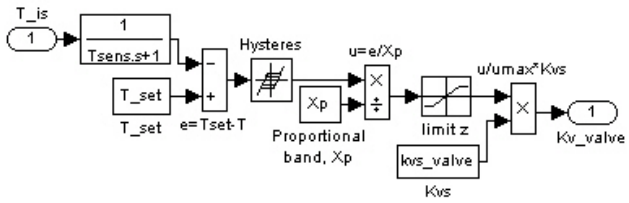


Figure 4 Thermostatic valve model

Since the valve is self-acting no actuator is needed, and as a consequence the main inertia within the controller is due to the time constant of the temperature sensor. Hjorthol [4] has shown that sensors placed directly in flowing water with an adequate accuracy can be described only by a single time constant. Due to internal friction in the valve there is always some degree of play/hysteresis within the valve. To reduce wear of the valve some hysteresis can also be introduced deliberately.

The proportional band, X_{prop}, is a measure of how large the control error, $e=T_{set} - T$, must be for the valve to fully open and is defined as:

$$X_{prop} = T_{z=0\%} - T_{z=100\%} \tag{9}$$

The valve opening degree, z, must be within the limits 0-100 %. Apart from this also some minimum opening degree > 0 % is sometimes introduced in order to facilitate the control. If an actuator had governed the valve position also the movement rate of the valve should be limited using a *"Rate limiter"*-block.

Once the valve position is known the corresponding k_v-value can be decided. In Figure 4 the valve is linear and the current k_v-value is obtained as the valve position times the k_v-value for a fully open valve. If the valve is not linear a look-up table containing the valve characteristics can be used. Once the k_v-value of the valve is known the flow can be calculated using (10) (see below).

Pumps used in DHW circulation systems are normally run at constant speed. The pump model used in the DHW circulation model is very simple and consists of a 1-D look-up table describing the pump characteristics. The input to the look-up table is calculated flow and the output is differential pressure across the pump. The interaction with the rest of the model then results in a new flow and so on. For a variable speed pump a 2-D look-up table can be used with the additional input of speed.

Calculation of flows in a DHW circulation system

In this section two methods of calculating flows in a DHW circulation system will be described. From this kind of systems water is frequently drawn off in the tapping-cocks. The size of the extracted flows is here regarded as known and is input to the calculations. Before the methods are described some general theory will be reviewed.

For fully turbulent flows the relationship between volume flow and differential pressure across a component can approximately be expressed as:

$$\dot{V} = k_v \cdot \sqrt{\Delta p} \tag{10}$$

Where (for linear valves): $k_v = k_{vs} \cdot z$

I.e., the volume flow is proportional to the square rot of the differential pressure and the proportionality factor is k_v. For valves the k_{vs}-value is frequently used to describe the capacity of the valve at 1 bar pressure drop. It should be noted that throughout this paper all components are defined by flow capacity, which should not be confused with flow resistance. By using (10) the equivalent k_v-value for N components connected in parallel/series can be calculated as, Persson [1]:

$$k_{v,eq,parallel} = \sum_{i=1}^{N} k_{v,i} \tag{11}$$

$$k_{v,eq,series} = \frac{1}{\sqrt{\sum_{i=1}^{N} \left(\frac{1}{k_{v,i}} \right)^2}} \tag{12}$$

Method for calculating flows based on an analogy to Kirchoff's law

Schematically the DHW circulation system as in Figure 1 could be regarded as:

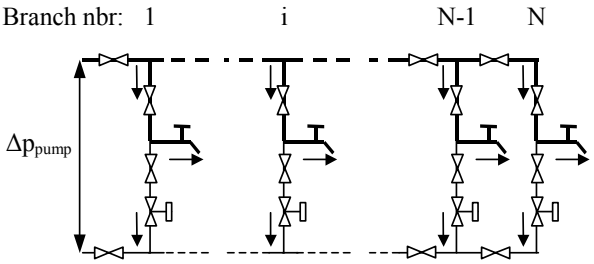


Figure 5 Schematic sketch of a DHW circulation system

Now study one of the branches (risers) in Figure 5. In detail each branch can be represented in the following way:

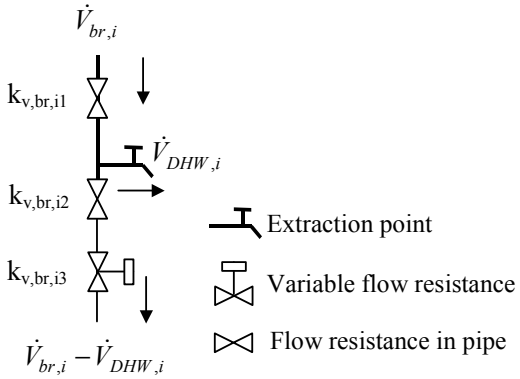


Figure 6 Details of one branch in Figure 5

The pressure drop across a branch, $\Delta p_{br,i}$, where flow is extracted in one point can be expressed as:

$$\Delta p_{br,i} = \Delta p_{br,i1} + \Delta p_{br,i2} + \Delta p_{br,i3} =$$

$$\left[\frac{\dot{V}_{br,i}}{k_{v,br,i1}}\right]^2 + \left[\frac{\dot{V}_{br,i} - \dot{V}_{DHW,i}}{k_{v,br,i2}}\right]^2 + \left[\frac{\dot{V}_{br,i} - \dot{V}_{DHW,i}}{k_{v,br,i3}}\right]^2 =$$

$$\left[\frac{\dot{V}_{br,i}}{k_{v,br,i1}}\right]^2 + \left[\frac{\dot{V}_{br,i} \cdot a_i}{k_{v,br,i2}}\right]^2 + \left[\frac{\dot{V}_{br,i} \cdot a_i}{k_{v,br,i3}}\right]^2 = \left[\frac{\dot{V}_{br,i}}{k_{v,br,i,eq}}\right]^2$$

From this we obtain:

$$k_{v,br,i,eq} = \frac{1}{\sqrt{\left[\frac{1}{k_{v,br,i1}}\right]^2 + \left[\frac{a_i}{k_{v,br,i2}}\right]^2 + \left[\frac{a_i}{k_{v,br,i3}}\right]^2}}$$

$$\text{Where: } a_i = \frac{\dot{V}_{br,i} - \dot{V}_{DHW,i}}{\dot{V}_{br,i}} \qquad (13)$$

Once the equivalent k_v-value for each branch, $k_{v,br,i,eq}$, is known the next step is to calculate the equivalent k_v-value for the whole system. The calculations start at the outer-most branch, cf. region *A* in Figure 7.

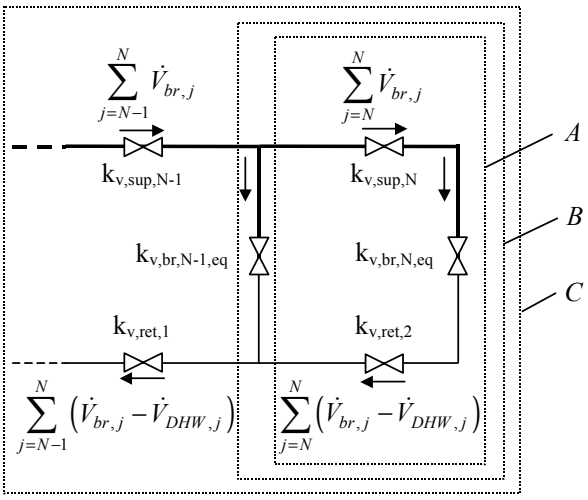


Figure 7 Calculation regions

The equivalent k_v-value for this region can be calculated according to (13). Note that the extracted flow in branch N has already been considered when calculating $k_{v,br,N,eq}$ and is not to be accounted for again in the term corresponding to the pressure drop in the branch. However, it should be accounted for in the term corresponding to the pressure drop in the return pipe. We obtain:

$$k_{v,eq,A} = \frac{1}{\sqrt{\left[\frac{1}{k_{v,sup,N}}\right]^2 + \left[\frac{1}{k_{v,br,N,eq}}\right]^2 + \left[\frac{c_N}{k_{v,ret,N}}\right]^2}}$$

$$\text{Where: } c_N = \frac{\sum_{j=N}^{N}\left(\dot{V}_{br,j} - \dot{V}_{DHW,j}\right)}{\sum_{j=N}^{N}\dot{V}_{br,j}} \qquad (14)$$

For region A, $c_N = a_N$, but otherwise $c_i \neq a_i$ if water is extracted in a branch j>i. Once $k_{v,eq,A}$ has been formulated the equivalent k_v-value for region B according to (11) becomes:

$$k_{v,eq,B} = k_{v,br,N-1,eq} + k_{v,eq,A} \qquad (15)$$

The next step is to calculate $k_{v,eq,C}$ and since this is a direct analogy to the formulation of $k_{v,eq,A}$ it is only stated that:

$$k_{v,eq,C} = \frac{1}{\sqrt{\left[\frac{1}{k_{v,\mathrm{sup},N-1}}\right]^2 + \left[\frac{1}{k_{v,eq,B}}\right]^2 + \left[\frac{c_{N-1}}{k_{v,\mathrm{ret},N-1}}\right]^2}}$$

$$\text{Where: } c_{N-1} = \frac{\sum_{j=N-1}^{N}\left(\dot{V}_{br,j} - \dot{V}_{DHW,j}\right)}{\sum_{j=N-1}^{N}\dot{V}_{br,j}} \qquad (16)$$

By performing corresponding calculations for the rest of the system until j=1, the equivalent k_v-value for the whole system, $k_{v,system}$, can be calculated. With an overall, known differential pressure for the system, Δp_{system}, the total flow is then obtained from (10).

As the total flow through the system is known, Δp across the first branch can be calculated. With known k_v-value for the first branch it is possible to calculate the flow through the branch and consequently the flow that continues further out in the system is also known. In this way the flow calculations propagate through the system until all flows are known.

It should be noticed that in the special case when no flow is extracted from the system, both a_i and c_i become unity. The equations then take the form of a direct analogy to Kirchoff's law, and all flows can be calculated explicitly. But as soon as a flow is extracted anywhere in the system, the equations have to be solved iteratively.

Calculating flows using the Hardy-Cross method

An alternative method of calculating the flow distribution in a DHW circulation system is to simply follow the flow through each branch. Starting at the beginning of the system and moving along the supply pipe, through a branch, back along the return pipe and finally across the pump the net pressure drop must be zero when returning to the point of origin. I.e.:

$$\Delta p_{sup} + \Delta p_{br} + \Delta p_{ret} - \Delta p_{pump} = 0 \qquad (17)$$

Using k_v-values and the notations in Figure 6 and Figure 7 this can for branch number *i* be expressed as:

$$\left\{\sum_{j=1}^{i}\left[\left(\sum_{k=j}^{N}\dot{V}_{br,k}\right)\Big/k_{v,sup,j}\right]^2\right\} + \left(\frac{\dot{V}_{br,i}}{k_{v,br,i1}}\right)^2 + \left(\frac{\dot{V}_{br,i} - \dot{V}_{DHW,i}}{k_{v,br,i2}}\right)^2 + \left(\frac{\dot{V}_{br,i} - \dot{V}_{DHW,i}}{k_{v,br,i3}}\right)^2 + \left\{\sum_{j=1}^{i}\left[\left(\sum_{k=j}^{N}\left(\dot{V}_{br,k} - \dot{V}_{DHW,k}\right)\right)\Big/k_{v,ret,j}\right]^2\right\} = \Delta p_{pump} \qquad (18)$$

By expressing (18) for branches 1 to N as many equations as unknown branch flows are obtained, making it possible to calculate iteratively the flows in each branch. The method is often called the Hardy-Cross method.

Regarding flow direction

In both methods described above the flow direction is regarded as known. In theory the extracted flow could be of such magnitude that the flow direction in the return pipe is changed. However, it can be shown that for correctly sized systems these situations are extremely rare and they are therefore disregarded in this work.

Comparison of flow models and how to implement them in Simulink®

The method based on Kirchoff's law and the Hardy-Cross method, as described above, from now on will be called the Kirchoff method and the Hardy-Cross method, respectively. Next will be described how to implement the models in Simulink®. Furthermore advantages and disadvantages with the two methods are discussed.

When water is extracted from the system both the Kirchoff and the Hardy-Cross method is iterative. In the case when no water is extracted the flow distribution can be calculated explicitly. Since the explicit calculations are much more rapid than the iterative ones it is important not to use an iterative method if not necessary. In Figure 8 is shown how an *"Enable"*-block can be used in Simulink® to switch between the iterative and explicit calculations. By checking if water is extracted or not and only using the iterative solver when needed the computational time can be reduced significantly.

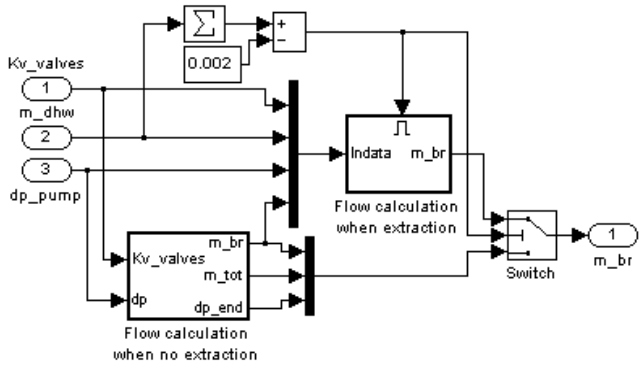


Figure 8 Method to switch between iterative and explicit flow calculation

If a system only consist of none-variable flow resistances it should be sufficient to calculate the flow distribution only once when a flow extraction starts or stops. This will reduce the computational time even further. The system modeled in this paper contains variable flow resistances (i.e. the thermostatic valves in Figure 1) that change often. For this reason there is little to be gained from only calculating the new flow distribution when there is a change in the system.

In Figure 8 the Kirchoff method is used when no water is extracted and otherwise the Hardy-Cross method is used. The Hardy-Cross method is not directly implemented in Simulink® and is instead formulated as a Matlab® function. Using the command "*fsolve"* in Matlab® the equation system according to (18) can be solved. An advantage with using "*fsolve"* is that it is possible to supply the iterative solver with an initial guess. An appropriate choice of initial guess each time there is a change in the system improves stability and decreases computational time. Here, the initial guess for the branch flows is chosen as the flow distribution when no water is extracted (supplied by the Kirchoff method, Figure 8) plus the extracted flow in each branch. It should be noticed that it would also be possible to implement the Hardy-Cross method directly in Simulink®.

When using the Hardy-Cross method implemented as a Matlab® function, the computational time can be decreased further by using the Matlab® Compiler command "*mcc"* and convert the Matlab® function to an S-function. This in addition makes it possible to use the Real-Time Workshop®, RTW, accelerator (which don't accept Matlab® functions). RTW normally greatly speeds up a Simulink® model, but due to the S-function the speed gain in this case is not as big when flow is extracted.

Despite the merits of the described methods to increase the simulation speed the fact remains that the Hardy-Cross method, when implemented as described, involves time-consuming calculations. The reason is the well-known fact that calling external functions in Simulink® considerably slows down the model. For this reason one should as far as possible use elementary blocks and not use Matlab® functions.

If instead the Kirchoff method is used also when flow is extracted from the system this could quite easily be implemented in Simulink®. Since the calculations are iterative, a so-called algebraic loop is obtained. RTW does not support algebraic loops and therefore the model cannot be accelerated. Still, since the flow calculation is performed within Simulink® no external application has to be called and the computational speed greatly exceeds the S-function calculations. From this vantage point the Kirchoff method appears very attractive, but it has one draw back: In Simulink® it is not possible to provide the iterative solver for the algebraic loop with an appropriate initial guess when there is a change in the system. If this would have been possible, the computational time could probably be decreased further, but above all such a feature could be expected to improve the stability of the solver. The latter fact would be desirable since, when solving the algebraic loop, some stability problems have in fact been observed. Since there is an increased risk of numerical problems the Kirchoff method, when implemented directly in Simulink®, should be used with some caution.

Computational time

In this part the computational times for some different types of models are investigated for two different situations. The purpose of the model is to calculate the flow distribution for the DHW circulation system in Figure 1. The system consists of 12 branches and 12 potential extraction points. The two situations studied is one case when no water is extracted from the system (no iterative calculations needed) and one case when water is extracted (iterative calculations needed). The period studied is 24 hours long and, in the case with extractions, flow is extracted during

45 % of the time. The flows are calculated using the models described in Table 1. Models 2 and 3 contain an algebraic loop and can therefore not be accelerated with Real-Time Workshop®. For models with both explicit and iterative solvers the latter is called only during extractions. The computational times for the two cases and the five different types of models are shown in Table 2. The computations were performed on an Intel Pentium 4 2.4 GHz with 736 MB RAM and Windows XP as operating system. Version 5.0 of Simulink® was used.

Model nbr	Description
1	Kirchoff method (explicit), implemented in Simulink®, RTW
2	Kirchoff method (iterative), implemented in Simulink®
3	Kirchoff method (explicit and iterative), implemented in Simulink®
4	Hardy-Cross method (iterative) and Kirchoff method (explicit), implemented as a S-function, RTW
5	Hardy-Cross method (iterative), implemented as an S-function, and Kirchoff method (explicit), implemented in Simulink®, RTW

Table 1 Description of models for calculating flow distribution

Model type	1	2	3	4	5
Comp. time no extractions [s]	0.8	822	2.8	25	1.0
Comp. time with extractions [s]	–	2000	1345	7279	7208

Table 2 Computational times for different cases

According to Table 2 the iterative models are time-consuming compared to the explicit models, compare 2 to 1 with no extractions. Also, it is more efficient to calculate flows within Simulink® than to call an S-function, compare 4 and 5 to 2 and 3 for the extraction case. It can further be noticed that the RTW accelerator may speed up a model greatly when it is possible to use this facility, compare 1 and 5 to 3 with no extractions.

Conclusions

In this paper it has been shown how the components of a DHW circulation system can be described in mathematical terms and implemented in Simulink®. When implementing the models elementary blocks should be used as far as possible. Also look-up-tables should be adopted instead of time consuming calculations within Simulink®. Two methods of calculating flow distribution in a system with extraction points are described. When implementing these methods, calculations should primarily be performed within Simulink® since this reduces the computational time. If numerical problems arise, an equation solver facility of Matlab® can be used since this enables the user to provide the solver with an initial guess. The equation solver should be called using an S-function. If possible, the RTW-accelerator should be used since it may greatly speed up calculations.

References

[1] Persson, T., *Tappvarmvattenreglering i fjärrvärmecentraler för småhus [Domestic hot water control in district heating substation for single family houses]*, Licentiate thesis, Department of Heat and Power Engineering, Lund University, Sweden, 2002

[2] Wollerstrand, J., Persson, T., *Proper functioning of domestic hot water circulation in residential buildings*, In: Renewable Energy, Proceedings of 17th International Conference on Efficiency, Costs, Optimization, Simulation and Environmental Impact of Energy and Process Systems, Mexico, 2004

[3] Gummérus, P.: *Analys av konventionella abonnentcentraler i fjärrvärmesystem [Analysis of conventional district heating substations in district heating systems]*, PhD thesis, Department of Energy Conversion, Chalmers University of Technology, Sweden, 1989, ISBN 91-7032-467-0

[4] Hjorthol, E. M.: *Optimization of design values in district heating substations by system simulation*, PhD thesis, Institutt for Varme-, Ventilasjons- og Sanitærteknikk, Norges Tekniske Høgskole, Trondheim, 1990, ISBN 82-7119-230-2

[5] Larsson, G., *Dynamik i fjärrvärmesystem* [*On dynamics in district heating systems]*, PhD thesis, Department of Thermo- and Fluid Dynamics, Chalmers University of Technology, Sweden, 1999

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

EVALUATION OF OPERATION CHARACTERISTICS OF THE COUNTER-FLOW COOLING TOWER – DESIGN AND OFF-DESIGN PERFORMANCE

Jee-Young Shin*[1], Young-Seok Son[1] and Byoung-Mu You[2]
[1]School of Mechanical Engineering, Dong-Eui University
[2]Dept. of Mechanical Engineering, Graduate School, Dong-Eui University
995 Eomgwangno, Busanjin-Gu, Busan, 614-714, Korea

ABSTRACT

The heat released from HVAC systems and/or industrial process should be rejected to the atmosphere. In the past, cooling water was supplied from tap water or river, and rejected to the sewerage or the river again. Recently, conventional methods cannot satisfy either economic criteria or environmental regulation because the cost of supply and discharge of cooling water is increasing tremendously, and the thermal pollution is regulated severely as well.

Cooling tower enhances its application due to the low power consumption and, especially, low water consumption down to 5% of the direct water- cooling system. Heat rejection is accomplished within the tower by heat and mass transfer between hot water droplets and ambient air.

Counter-flow type cooling tower dominates the Korean market, and is widely used in the petrochemical industry, iron industry, and HVAC plant. However, the design of a cooling tower depends on the existing data and/or the procedure is lack of consistency. Design and off-design performance analysis has not been completed yet, which is one of the key parameters in the cooling tower performance evaluation. In this study, existing theories on cooling tower design were reviewed and summarized.

A program which computerizes the design procedure has been completed to keep consistency in the design. The off-design performance analysis program has been developed to analyze easily the performance characteristics of a counter-flow type cooling tower with various operating conditions. Through the experiments on various operating conditions, the off-design program has been verified. Finally, the operation characteristics with various operating conditions were evaluated by using this program.

Keywords: Counter-flow cooling tower, Operation characteristics, Design condition, Off-design performance.

[1]

1. INTRODUCTION

The heat released from HVAC systems and/ or industrial process should be rejected to the atmosphere. For example, cooling media such as water are often used to remove heat from condenser or heat exchanger of the energy system. In the past, cooling water was supplied from tap water or river, and rejected to the sewerage or the river again. Recently, conventional methods cannot satisfy either economic criteria or environmental regulation because the cost of supply and disemboguement of cooling water is increasing tremendously, and the thermal pollution is regulated severely as well.

Air-cooled heat exchanger can be an alternative, but it requires high initial investment cost and high fan power consumption. Cooling tower enhances its application due to the low power consumption and, especially, low water consumption down to 5% of the direct water-cooling system. Heat rejection is accomplished

[1] *Corresponding author. Phone:+82-51-890-1650, Fax:+82-51-890-2232, E-mail:jyshin@deu.ac.kr

within the tower by heat and mass transfer between hot water droplets and ambient air.
The operation theory of cooling tower was suggested by Walker in 1923, however, the generally accepted concept of cooling tower performance was developed by Merkel in 1925. A simplified Merkel theory has been used for the analysis of cooling tower performance and Lichtenstein introduced a graphical method. Baker and Shryock tried to minimize the error due to the assumptions of Merkel theory [1].
ASHRAE developed the cooling tower performance curves based on the Merkel theory in 1975, and several researchers have paid attention to the cooling tower performance by numerical analysis or experiments on the fluid flow phenomena and combined heat and mass transfer in cooling tower [2-6].
Counterflow type cooling tower dominates the Korean market, and is widely used in the petrochemical industry, iron industry, and HVAC plant. However, the design of a cooling tower depends on the existing foreign data and/or the procedure is lack of consistency. Design and off-design performance analysis has not been completed yet, which is one of the key parameters in the cooling tower performance evaluation.
In this study, existing theories on cooling tower design were reviewed and summarized. Previous design methodology of the company is thought to be lack of the theoretical background in choosing the characteristic curve of the tower and, thus, the design point. In this sense, a program which computerizes the design procedure has been completed to keep consistency in the design.
The off-design performance analysis program has been developed to analyze easily the performance characteristics of a counterflow type cooling tower with various operating conditions. Through the experiment on various operating conditions, the off-design program has been verified.

2. BASIC THEORY

2.1 Merkel equation

Heat transfer rate in the cooling tower is represented by the difference between the enthalpy of moist air at bulk water temperature and the enthalpy of moist air.
Merkel equation describes the heat transfer characteristics of a filler at the design condition. It needs several assumptions: (1)effect of evaporation does not exist, (2)thermal and mass diffusion coefficients of air/water system are the same. The analysis combines the sensible and latent heat transfer between air and water droplets in the tower.
Total heat transfer rate per unit volume of a filler(dV) from the interface to the air is the sum of sensible heat(dq_S) and latent heat(dq_L).

$$dq_S = U_G a dV\left(T'' - T\right) \tag{1}$$

$$dq_L = h_{fg} dm = h_{fg} K' a dV\left(W'' - W\right) \tag{2}$$

Energy conservation principle with the assumption that the interface temperature is same as the air temperature derives the following equation.

$$Lc_{pw} dt = KadV\left(h' - h\right) \tag{3}$$

Integration of Eq.(3) results in Eq.(4).

$$\frac{KaV}{L} = \int_{t_2}^{t_1} \frac{c_{pw} dt}{\left(h' - h\right)} \tag{4}$$

2.2 Counterflow type cooling tower

Left-hand side of Eq.(4) is a dimensionless parameter called NTU(number of transfer unit) which is the characteristic value of the filler and represents the heat transfer capacity, that is, the required heat transfer area. It is a function of air and water temperature, independent of the size of the tower or the shape of the filler. Counterflow cooling diagram shown in Fig.1 is convenient to integrate Eq. (4). The curves indicate the drop in water temperature (Point A to Point B). The temperature difference between the water entering and leaving the cooling tower(A-B) is the range. The difference between the leaving water temperature and the entering air wet-bulb temperature(B- C) is the approach of the cooling tower.

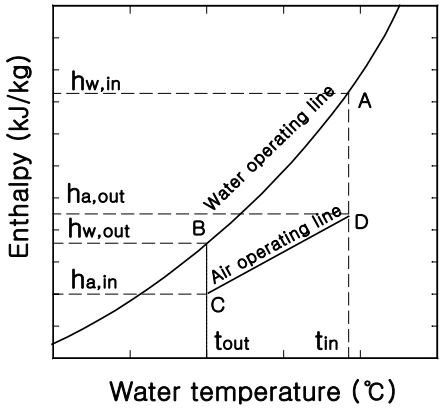


Fig. 1 Enthalpy-temperature diagram of air and water

The equations are not self-sufficient, therefore, Tchebycheff integration is applied to get the approximate value.

$$\int_{t_1}^{t_2} \frac{c_{pw}dt}{h'-h} \approx c_{pw}(t_1 - t_2) \times \frac{1}{4}\sum_{i=0}^{4}\left(\frac{1}{h_i' - h_i}\right) \qquad (5)$$

where is the air enthalpy at the interface with bulk water temperature, and means the air enthalpy with the air stream temperature.
The enthalpy of point C and D in Fig.1 can be represented as Eq.(6) based on the energy balance.

$$h_2 = h_1 + c_{pw}\frac{L}{G}(t_2 - t_1) \qquad (6)$$

The slope of the air operating line CD equals *L/G*, the ratio of the water flow rate to the air flow rate.
Packing characteristic curve represents the heat transfer characteristics of the filler, which is shown in Fig.2. It is a typical correlation of the performance characteristic of a cooling tower showing the variation of available *KaV/L* with *L/G* for a constant air velocity on logarithmic coordinates. If the air flow rate decreases with constant water flow, the heat transfer at the filler will diminish as in Fig.2.
Packing characteristic curve represents the heat transfer characteristics of the filler, which is shown

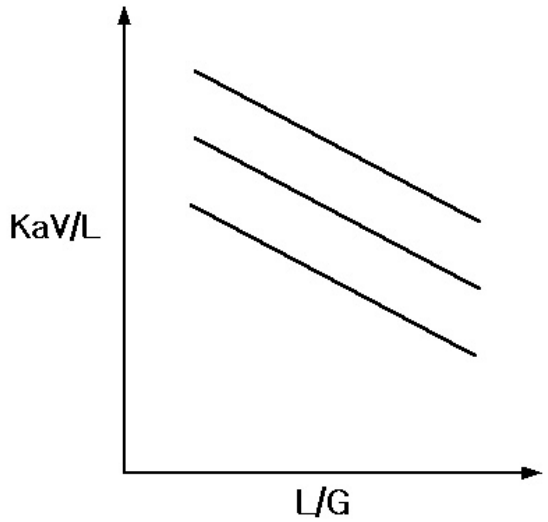


Fig. 2 Characteristic curve of a cooling tower

in Fig.2. It is a typical correlation of the performance characteristic of a cooling tower showing the variation of available *KaV/L* with *L/G* for a constant air velocity on logarithmic coordinates. If the air flow rate decreases with constant water flow, the heat transfer at the filler will diminish as in Fig.2.
The cooling tower characteristic curve in Fig. 2 corresponds to the following relation from the experimental results [7].

$$\frac{KaV}{L} = c\left(\frac{L}{A}\right)^m \left(\frac{G}{A}\right)^n \qquad (7)$$

where *A* is the frontal area, and *c, m,* and *n* are experimental constants.

3. DESIGN AND OFF-DESIGN ANALYSIS

3.1 Design analysis

The design procedure of a cooling tower is as follows;
(1)Input design conditions: water inlet & outlet temperature, water flow rate, air inlet temperature.
(2)Assume the ratio *L/G*, and evaluate the exit air enthalpy using Eq.(6).
(3)Calculate required NTU from Eq.(4).
(4)Calculate characteristic NTU from Eq.(7).
(5)Iterate until the required NTU equals the characteristic NTU.

(6)Set design NTU.

3.2 Off-design analysis

Off-design performance analysis is required to check whether the equipment operate normally. Because the equipment does not always operate on the design condition in the field, the off-design performance analysis is inevitable to the efficient and energy saving operation.

The procedure of the off-design analysis of a cooling tower is as follows;

(1)Evaluate the ratio *L/G* at the off-design condition.

(2)Evaluate the exit air enthalpy by assuming inlet water temperature.

(3)Calculate outlet water temperature and approach.

(4)Calculate required NTU using Eq.(4).

(5)Evaluate characteristic NTU with characteristic curve of design stage.

(6)Iterate until the required NTU equals the characteristic NTU.

Table 1 Design data (base case)

Item	Unit	
Circulation water flow rate	m^3/hr	3.9
Hot(inlet) water temp.	^oC	37
Cold(outlet) water temp.	^oC	32
Inlet air wet bulb temp.	^oC	27
Relative humidity	%	60
Nominal capacity	kcal/hr	19,500

(7)Set corresponding *L/G* and characteristic NTU.

3.3 Verification

Field test was run to verify the off-design performance analysis procedure. The temperature of the water reservoir is raised to a setting value. The heated water is drawn into the tower and cooled by air. Flow rate and inlet & outlet temperature of water, and temperature and humidity of inlet air are measured.

The design data used in the test are summarized in Table 1. The capacity of the tower is

Table 2 Off-design performance data (experiment and prediction)

	Design condition	Case I	Case II	Case III	Case IV
Water flow rate(x10^3 m^3/s)	0.9	3.0	2.98	3.02	1.1
Inlet air dry bulb temp.(^oC)	32.8	27.2	29.3	25.4	26.7
Relative humidity(%)	70	64	40.3	34.7	50
Inlet air wet bulb temp.(^oC)	28.1	22	19.6	15.6	19.3
Hot water temp.(^oC)	40	38	39	30	34.5
Cold water temp.(Exp., ^oC)	34	34.5	35.4	28.5	30.4
Cold water temp.(Cal., ^oC)	34	36.1	35.8	27.8	30.9

Table 3 Comparison of the design data (current calculation, CTI Bluebook, company's own design)

Item	Unit	Case I	Case II	Case III	Case IV	Case V
Circulation water flow rate	m^3/hr	680	600	772.5	800	220
Hot water temp.	^oC	53	45	43	37	43
Cold water temp.	^oC	35	35	32	32	32
Wet bulb temp.	^oC	27	28	28	27	28
Cell quantity	ea	1	5	3	2	2
$(L/G)_{CAL}/(L/G)_{CTI}$		0.975	1.033	0.932	0.993	.932
$(NTU)_{CAL}/(NTU)_{CTI}$		0.969	0.944	1.006	0.962	1.006
$(L/G)_{CAL}/(L/G)_{company}$		1.625	1.188	1.186	1.15	1.717
$(NTU)_{CAL}/(NTU)_{company}$		1.409	1.245	1.275	1.268	1.53

19,500kcal/hr, which is designed by the DHTech corporation, Korea(Model No. DCT-5R).
Both measured and calculated outlet water temperature are represented in Table 2, and it shows that the off-design analysis predicts the performance fairly well. By comparing the measured and predicted values, the design and off- design analysis procedure is verified.
Existing design data and the results of this study are compared in Table 3. Existing design data, hand calculated data using CTI Bluebook, and data from this analysis program are shown. CTI Bluebook is the handbook for the design of cooling tower, say, NTU by CTI(Cooling Tower Institute) of U.S.A.
Results of this study are well consistent with the data of CTI Bluebook, and also predict the existing data with moderate error. The existing data are relatively subjective because the selection of the characteristic curve is dominated by the design engineer. Case I and V of the existing data are far away from the CTI Bluebook and this study. This seems to be due to the engineer's choice on the factors irrelevant to the thermal performance of a cooling tower such as the size of the tower and fan consumption.
The selection of the characteristic curve in Fig.2 changes everything such as thermal performance, cost, and power consumption. This is the reason why the design procedure should be computerized to prevent the engineer's subjective criteria.

3.4 Off-design performance of a cooling tower
It is very difficult to maintain the actual operating conditions of a cooling tower at the design condition. Also the off-design performance on the various operating conditions should be provided to the customer. The importance of an off-design performance analysis cannot be overemphasized.
Through the validation procedure of the current study, the performance characteristic of a cooling tower with various operating conditions is reviewed in the following sections. The design data in Table 1 are used as the reference condition.

3.4.1 Variation of wet-bulb temperature
The influence of a wet-bulb temperature on the performance of a cooling tower is studied under constant water flow, air flow, and water inlet

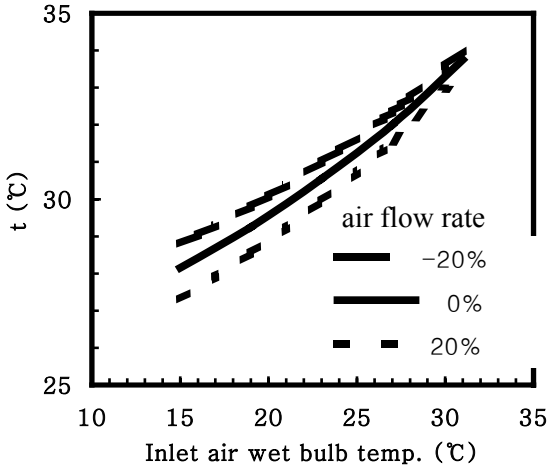


Fig. 3 Effect of wet bulb temperature on the exit water temperature

temperature. Water outlet temperature as a tower performance is represented in Fig.3. Also air flow rate is changed by ±20% from the design point.
Even with the wide range of wet bulb temperature change more than 15℃, water temperature varies within a relatively small range, 5 ~ 6 ℃. Considering that the heat capacity of water is much higher than that of air in cooling tower operation, it can be predicted that the effect of a wet-bulb temperature on the water temperature is not so critical.
Water outlet temperature decreases and the effect of the wet-bulb temperature becomes more sensitive with increasing air flow. When the wet-bulb temperature is too high, little effect on the performance is seen even with 20% increase of air flow.
These trend will be different for each cooling tower and each operating condition, respectively, so it should be suggested to the customer that the off-design analysis be done to predict the extent of change. For example, it could be suggested that increasing air flow be not helpful on the performance if the wet-bulb temperature of inlet air is above a threshold value.

3.4.2 Variation of range of water temperature
Cooling performance with range of water is shown in Fig.4. Larger range means higher water outlet temperature. This means that increasing

cooling load can be met by the larger range for a given heat exchanger. If the range is changed, the user has to set the water inlet temperature to a different value for the required performance. Water outlet temperature decreases with increasing air flow.

3.4.3 Variation of water flow rate

The effect of a water flow on the performance is represented in Fig.5. Even with ±20% variation of water flow, the outlet water temperature changes within 1℃.

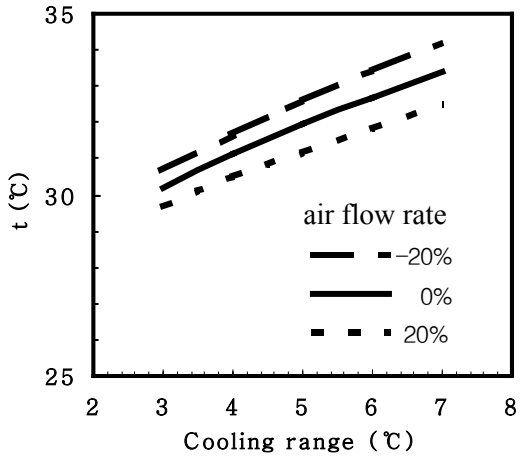


Fig. 4 Effect of cooling range on the exit water temperature

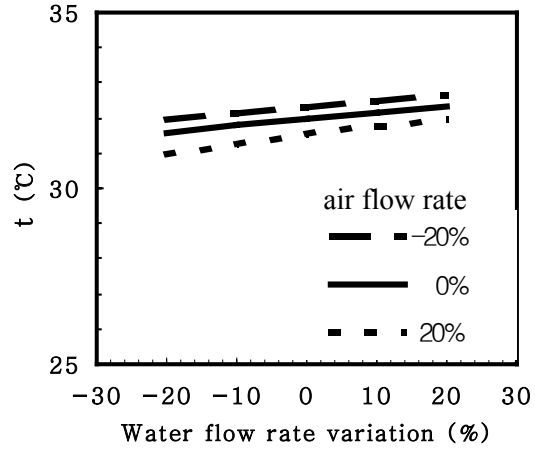


Fig. 5 Effect of water circulation rate on the exit water temperature

Cooling capacity becomes higher with increasing air flow, but it does not practically change when the water flow rate increases about 20%.

Sensitivity of such parameters on the cooling tower performance will be different for each cooling tower, therefore off-design analysis is inevitable and the feedback of actual performance data from the field is also necessary for the setup of a characteristic curve of a cooling tower.

4. SUMMARY

(1)Cooling tower design procedure is computerized and the results are compared with existing design data. The procedure is necessary to prevent the subjective judgement of a design engineer and the appropriate selection of a characteristic curve is also important.

(2)Off-design analysis is setup and verified. This can be used for the evaluation of a cooling tower performance on every different operating conditions.

(3)The effects of some parameters such as wet-bulb temperature, water flow, range of water temperature are studied by off-design analysis. The changes of inlet air wet-bulb temperature and range of water temperature affect the cooling tower performance considerably but the effect of water flow on the performance is not so high. Off-design performance should be carefully reviewed at the design stage because it will be different for every equipment and operating condition.

NOMENCLATURE

A : frontal area [m2]
a : area of water interface per unit volume [m^2/m^3]
c_p : specific heat [kJ/kg ℃]
G : mass flow rate of air [kg/s]
H : enthalpy [kJ/kg]
K : overall mass transfer coefficient [kg/s m^2]
L : mass flow rate of water [kg/s]
M : mass [kg]
q : heat transfer rate [kJ/s]
T : dry bulb temperature [℃]
t : water temperature [℃]
U : overall heat transfer coefficient [kW/m^2K]
V : cooling tower volume [m^3]

W : absolute humidity

Superscripts

′ : at bulk water temperature

″ : at interface

Subscripts

a : air

G : between interface and air

L : latent heat

S : sensible heat

w : water

1,2: inlet and outlet of cooling tower

ACKNOWLEDGMENTS

This study was supported by Korea Institute of Science & Technology Evaluation and Planning(KISTEP) and Ministry of Science & Technology, Korean Government through its National Nuclear Technology Program(Project number:M20362000001-03B0500-02310).

REFERENCES

[1] Baker DR, Shryock HA. *A comprehensive approach to the analysis of cooling tower performance.* J. of Heat Transfer 1961; August: 339-350.

[2] Bernier MA. *Cooling tower performance: Theory and experiments.* ASHRAE Transactions 1994; 3794: 114-121.

[3] Webb RL, Villacres, A. *Cooling tower performance.* ASHRAE Journal 1984; November: 34-40.

[4] Lee HC, Bang KH, Kim MH., *Experimental study on the thermal performance of a cooling tower.* Korean Journal of Air-Conditioning and Refrigeration Engineering 1998; 10(1):88-94.

[5] Lee KH, Choi, WY, So HY. *Comparison of performance of cooling tower with various shape of packings*. Korean Journal of Air-Conditioning and Refrigeration Engineering 1998; 10(3):358-367.

[6] Kim YS, Seo MK, Lee SK. *A study on the counter-flow cooling tower performance analysis using NTU- method*. Korean Journal of Air-Conditioning and Refrigeration Engineering 1999;11(5):598-604.

[7] ASHRAE. *ASHRAE handbook-HVAC systems and equipmen(Ch.37)*, Atlanta: American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc., 1996

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

ANALYSIS ON THE PERFORMANCE OF SHELL-AND-TUBE HEAT EXCHANGERS WITH CONVENTIONAL AND SPIRAL BAFFLE PLATES

Young-Seok Son*[1], Sung-Kwon Kim[2] and Jee-Young Shin[1]
[1]School of Mechanical Engineering, Dong-Eui University
[2]Dept. of Mechanical Engineering, Graduate School, Dong-Eui University
995 Eomgwangno, Busanjin-Gu, Busan, 614-714, Korea

ABSTRACT

In a conventional shell-and-tube heat exchanger with vertical baffle plates, fluid contacts with tubes flowing up and down in a shell, therefore heat transfer is deteriorated due to the stagnation portions occurring near the contact regions of the shell and baffles. It is, therefore, necessary to improve heat exchanger performance by changing fluid flow in the shell. In this study, a shell-and-tube heat exchanger with spiral baffle plates is considered to improve the performance of the conventional shell-and-tube heat exchanger. In this type of heat exchanger, fluid contacts with tubes flowing rotationally in the shell. Therefore, it could improve heat exchanger performance considerably because stagnation regions in the conventional shell-and-tube heat exchanger could be eliminated. The shell-and-tube heat exchangers with conventional and spiral baffle plates are simulated three-dimensionally using a commercial thermal-fluid analysis code, CFX4.2, and the results are compared. It is proved that the shell-and-tube heat exchanger with spiral baffle plates is superior to the conventional heat exchanger in terms of heat transfer.

Keywords: Heat exchanger performance, Shell-and-tube heat exchanger, Conventional baffle plate, Spiral baffle plate.

1. INTRODUCTION

Heat exchangers are devices that provide the flow of thermal energy between two or more fluids at different temperatures. Heat exchangers are used in a wide variety of applications such as power production, process and chemical industries, cooling of electronic systems, environmental engineering, waste heat recovery, manufacturing industry, air conditioning, and refrigeration.

There are many types of heat exchangers. A shell-and-tube heat exchanger is built of round tubes mounted in a cylindrical shell with the tubes parallel to the shell. One fluid flows inside the tubes, while the other fluid flows across and along the axis of the shell. Shell-and-tube heat exchangers offer great flexibility to meet almost any service requirements. They can be designed for high pressure difference between fluid streams. They are widely used in process industries, in power plants as condensers, steam generators, and feed water heaters, and in air conditioning and refrigeration systems [1, 2].

The thermal and flow analyses of heat exchangers have been performed [3, 4]. The flow and heat transfer characteristics of a conventional shell-and-tube heat exchanger have been studied experimentally [5-8] and numerically [9, 10].

In shell-and-tube heat exchangers, vertical baffles are installed in the shell side to increase the convection heat transfer coefficient of the shell-side fluid by inducing turbulence and a cross-flow velocity component. Also baffles support tubes for structural rigidity, preventing tube vibration and sagging. However, the stagnation area occurring near the contact regions of the shell and baffles in a conventional shell-and-tube heat exchanger has an unfavorable effect on the heat transfer between the shell and tube side fluids. Fins are attached to the tubes in order to increase heat transfer efficiency,

*Corresponding author. Phone:+82-51-890-1648, Fax:+82-51-890-2232, E-mail:ysson@deu.ac.kr

but there exists a limit in the enhancement of heat exchanger performance.

The efforts to improve the shell-side flow characteristics are made using the spiral baffle plates instead of vertical baffle plates in a conventional shell-and-tube heat exchanger. The new design of the shell-and-tube heat exchanger with spiral baffle plates is proposed to eliminate the stagnation area in the shell-side flow in a conventional shell-and-tube heat exchanger. Rotational flow in the shell side caused by the spiral baffle plates eliminates the stagnation region and improves heat transfer significantly.

In this paper, three-dimensional numerical analyses are performed for the shell-and-tube heat exchangers with conventional and spiral baffle plates using the commercial thermal-fluid analysis code, CFX4.2 [11]. Shell and tube side flow fields, pressure drops, and heat transfer characteristics in the heat exchangers are analyzed. The results of the shell-and-tube heat exchanger with spiral baffle plates are compared with those of the conventional shell-and-tube heat exchanger.

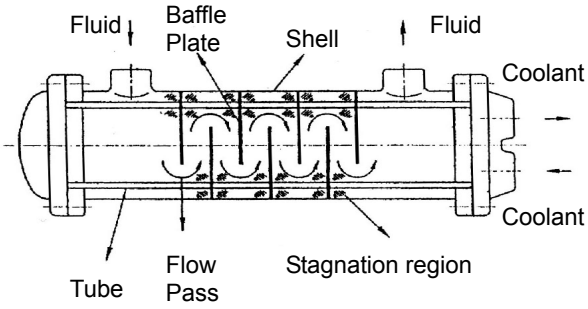


Figure 1: Conventional shell-and-tube heat exchanger with vertical baffle plates

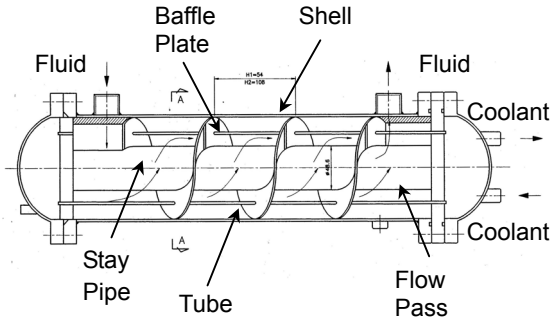


Figure 2: Shell-and-tube heat exchanger with spiral baffle plates

2. NUMERICAL ANALYSIS

2.1 Numerical model

Shell-and-tube heat exchangers are built of round tubes mounted in a large cylindrical shell with the tube axis parallel to that of the shell. In a conventional shell-and-tube heat exchanger with vertical baffles, the shell-side stream flows across pairs of baffles, and then flows parallel to the tubes as it flows from one baffle compartment to the next as shown in Fig. 1. The stagnation region occurring in a conventional shell-and-tube heat exchanger has an unfavorable effect on the heat transfer between the shell and tube side fluids.

The spiral baffle plate is one of the methods to improve the shell-side flow characteristics. The baffle has a spiral shape like a screw and does not have baffle cut as depicted in Fig. 2. Flow in the shell side rotates circumferentially along the spiral baffle plate, and it contributes to the elimination of the stagnation region and enhancement of the heat transfer significantly.

Fig. 3 shows the basic model with 7 spiral baffle plates of this study. The inter-baffle spacing is 72 mm, and the physical dimensions are same as those of the conventional heat exchanger, shell diameter and axial length of 114.3 mm and 667 mm, tube diameter and thickness of 28.4 mm and 2 mm. One

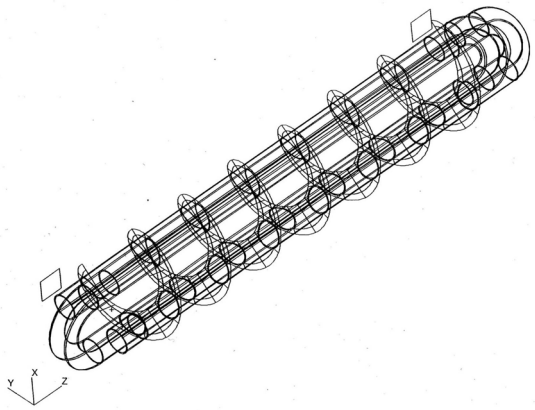


Figure 3: Schematic internal view of the shell-and-tube heat exchanger with spiral baffle plates

shell pass and four tube passes system is assumed, and there is one tube per tube pass.

2.2 Governing equations and boundary conditions

The model of CFX4.2 uses conservation of mass, momentum, and energy principles.

The continuity equation is

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \left(\rho \vec{U}\right) = 0 \tag{1}$$

The general momentum equation is

$$\frac{\partial \rho \vec{U}}{\partial t} + \nabla \cdot \left(\rho \vec{U} \times \vec{U}\right) = -\vec{B} + \nabla \cdot \sigma \tag{2}$$

where σ is the stress tensor as in equation (3).

$$\sigma = -P\delta + \left(\xi - \frac{2}{3}\mu\right)\nabla \cdot \vec{U}\delta + \mu\left(\nabla \vec{U} + \left(\nabla \vec{U}\right)^T\right) \tag{3}$$

where ρ is the fluid density, $\vec{U}$ is the fluid velocity, t is the time, and $\vec{B}$ is the body force. P is the pressure, μ is the viscosity, and ξ is the bulk viscosity. The k-ε turbulence model is included.

The energy equation is

$$\frac{\partial \rho H}{\partial t} + \nabla \cdot \left(\rho \vec{U} H\right) - \nabla \cdot \left(k \nabla T\right) = \frac{\partial P}{\partial t} \tag{4}$$

where H is the total enthalpy(static enthalpy plus kinetic energy) as shown in equation (5), h is the static enthalpy, k is the thermal conductivity, and T is the temperature.

$$H = h + \frac{1}{2}\vec{U}^2 \tag{5}$$

Thermodynamic equations of state are added to solve these sets of equations as $\rho = \rho(T, P)$, $h = h(T, P)$. Heat conduction equation for solid regions is as follows.

$$\frac{\partial}{\partial t}\left(\rho_s H\right) - \nabla \cdot \left(k_s \cdot \nabla T\right) = 0 \tag{6}$$

For the boundary conditions, uniform velocity and temperature are given at the inlet boundaries of tube and shell. Pressures are given at the outlet boundaries, and the gradients of other variables are set to zero. Inlet velocity for turbulent case is set as ten times higher than that of laminar case. Boundary conditions are summarized in Table 1, and water flows in both tube and shell sides.

2.3 Numerical method

CFX4.2 uses finite volume method [12], and the SIMPLEC algorithm [13] is employed. The discretized equations are under-relaxed using the relaxation factors 0.7 and 0.65 for energy and momentum equation, respectively. Orthogonal coordinate system and body fitted grid are used for computation. The grid shape is also important in the accuracy of the solution. CFX4.2 employs multi-block grid structures and uses elliptic smoothing that provides smooth meshes to ensure good grids of smoothness and near orthogonality [11]. During computation, it is assumed that the convergence is reached when the error in continuity, the mass source residual which is defined as equation (7), has fallen below 0.1.

$$R = \sum_{all\ domain} \left| \sum a_{nb} \Phi_{nb} + b - a_p \Phi_p \right| \tag{7}$$

Table 1: Boundary conditions

		Unit	Laminar	Turbulent
Tube inlet	*U,V*	m/s	0	0
	W	m/s	0.07893	0.7893
	T	K	303.15	303.15
Tube outlet	*P*	Pa	0	0
Shell inlet	*U, W*	m/s	0	0
	V	m/s	- 0.03316	- 0.3316
	T	K	373.15	373.15
Shell outlet	*P*	Pa	0	0

Table 2: Summary of grid independence study

No. of cells	No. of iteration	Computation time (days)	Mass balance (kg/s)			Energy balance (W)		
			Inlet	Outlet	Error (%)	Inlet	Outlet	Error (%)
166,976	600	7	0.5643	0.5643	0.0	94855	93717	1.2
214,208	1400	18	0.3069	0.3069	0.0	75370	74710	0.8
276,540	1000	21	0.3070	0.3070	0.0	75370	71080	5.6

3. RESULTS

To select appropriate number of cells guaranteeing accuracy and efficient computation time, grid independence study has been performed. Table 2 shows the effects of the number of cells on the results. As seen in the table, the coarse grid provides small error in the mass and enthalpy balance at the inlet and outlet, however, the difference in the absolute values between the coarse grid and the intermediate grid is so significant. Fine grid shows relatively large error in enthalpy balance because the number of iteration was limited considering the excessive computation time. Intermediate grid offers nearly same values as those of fine grid, and relatively small error in the enthalpy balance. Therefore, among the coarse, intermediate, and fine grids, intermediate grid was selected considering both the accuracy of the solution and the computation time.

Figure 4 shows velocity vectors on a transverse plane at X = 0.023 m for (a) the whole plane and (b) Z = 0.237 ~ 0.478 m plane in a conventional shell-and-tube heat exchanger. This plane shows the flow fields in the two tubes and the shell, which is moved by 0.023 m from the center in the radial direction. As seen in the figure, the tube-side fluid flows at nearly uniform velocity. The shell-side fluid flows up and down along the baffle plates. However, it flows parallel to the tubes within a baffle window, because the resistance is lower in the axial direction compared with the cross-flow direction. Stagnation regions occur near the contact regions of the shell and baffle plates, which could partially explain the ineffectiveness of heat transfer in the conventional heat exchanger.

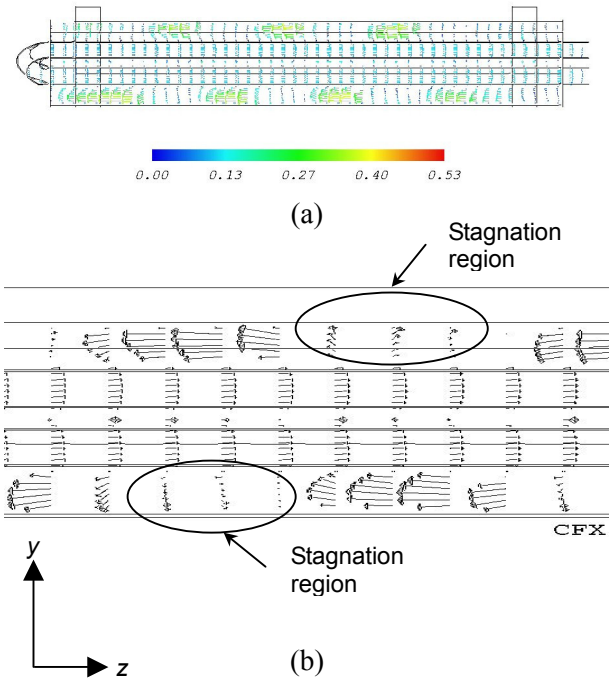


Figure 4: Velocity vectors on a transverse plane at X = 0.023 m for (a) the whole plane and (b) Z = 0.237 ~ 0.478 m plane – with conventional baffle

Figure 5 shows velocity vectors on a transverse plane at X = 0.037 m for (a) the whole plane and (b) Z = 0.241 ~ 0.435 m plane in a shell-and-tube heat exchanger with spiral baffle plates. This plane shows the flow fields in the one tube and the shell. The shell-side fluid flows along the spiral baffle plates accompanying rotation. The magnitude of the velocity vector is seen very small in the figure because only the component in this plane could be seen. However, it is also the evidence of the rotation of the flow field itself along the spiral baffle plates. Stagnation region seems to be reduced evidently as compared with the conventional shell-and-tube heat exchanger.

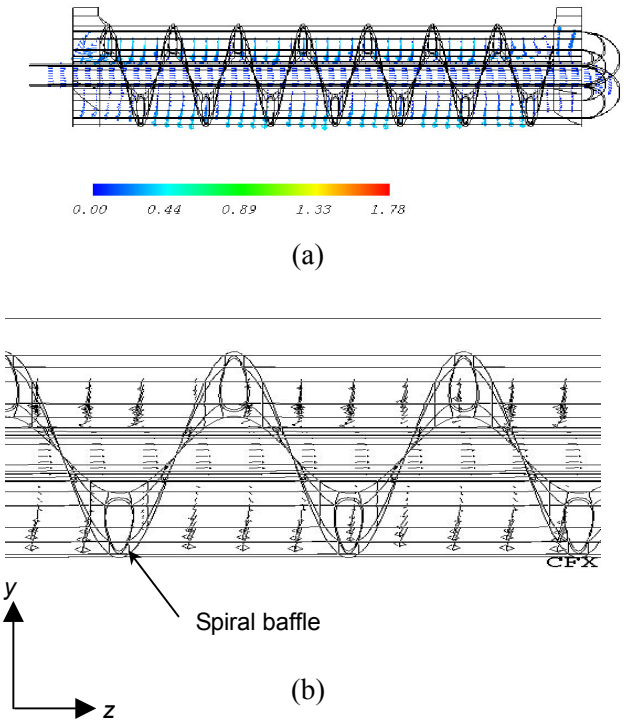


Figure 5: Velocity vectors on a transverse plane at X = 0.037 m for (a) the whole plane and (b) Z = 0.241 ~ 0.435 m plane – with spiral baffle

Figure 6 shows velocity vectors on a vertical plane at Z = 0.15 m in a conventional shell-and-tube heat exchanger. Again we could observe that the shell-

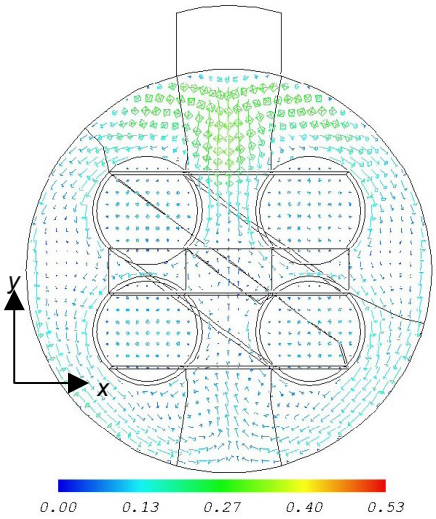


Figure 6: Velocity vectors on a cross-sectional plane at $Z = 0.15$ m – with conventional baffle

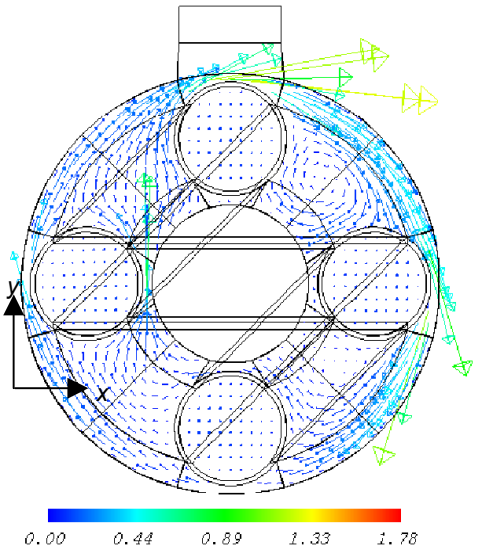


Figure 7: Velocity vectors on a cross-sectional plane at $Z = 0.19$ m – with spiral baffle

side fluid flows up and down along the baffles. The flow field at the bottom of the shell is quite stable in this cross-section, so the heat transfer enhancement is not substantial.

For the shell-and-tube heat exchanger with spiral baffle plates, the velocity vectors on a vertical plane at $Z = 0.19$ m are shown in Fig. 7. The shell-side fluid flows rotationally along the spiral baffles, and the fluid accelerates at the narrow space between the tube and the shell. The flow vorticities occurring at the shell side between the tubes increase the heat transfer between the tube and the shell.

Table 3 summarizes the analyses results for the shell-and-tube heat exchangers with conventional and spiral baffle plates. The sizes of tube and shell and the boundary conditions are all the same. The number of baffles is 7 for both cases. The overall

Table 3: Comparison of the performance of the shell-and-tube heat exchanger with conventional and spiral baffle plates

	Type of baffle	Conventional		Spiral	
	Flow	Laminar	Turbulent	Laminar	Turbulent
Shell side	Inlet temp.($T_{h,i}$), K	373.15	373.15	373.15	373.15
	Outlet temp.($T_{h,o}$), K	369.55	369.77	369.11	368.64
	Inlet pressure($P_{h,i}$), Pa	669	67900	9880	980000
	Outlet pressure($P_{h.,o}$), Pa	0	0	0	0
	Temp. drop(ΔT_h), K	3.60	3.38	4.04	4.51
	Pressure drop(ΔP_h), Pa	669	67900	9880	980000
Tube side	Inlet temp.($T_{c,i}$), K	303.15	303.15	303.15	303.15
	Outlet temp.($T_{c,o}$), K	325.90	324.13	324.30	326.76
	Inlet pressure($P_{c,i}$), Pa	21.2	1630	35.2	3050
	Outlet pressure($P_{c.,o}$), Pa	0	0	0	0
	Temp. rise(ΔT_c), K	22.75	20.98	21.15	23.61
	Pressure drop(ΔP_c), Pa	21.2	1630	35.2	3050
Heat transfer rate (Q), W		4680	43163	4351	48569
Log mean temperature diff.(ΔT_{lm}), K		56.28	57.37	56.76	55.39
Overall heat transfer coeff.(U), W/m^{2}K		348.0	3148.0	379.5	4340.9

heat transfer coefficient is calculated as follows.

$$U = Q / A\Delta T_{lm} \quad (8)$$

where Q is heat transfer rate, ΔT_{lm} is log mean temperature difference, and A is the surface area of the tubes.

As seen in the table, heat transfer is enhanced for the heat exchanger with spiral baffle plates compared with the conventional heat exchanger. This is explained by elimination of the stagnation region in the shell side due to the spiral baffle plates causing rotational flow in the shell side. The 9% higher overall heat transfer coefficient for laminar flow, and 38% for turbulent flow are observed with spiral baffle plates. Tube-side pressure drops are negligibly small for both cases, however, shell-side pressure drop of spiral baffle case is greater than that of conventional baffle case due to the increased shell-side fluid velocity. At the present model, heat transfer enhancement for the heat exchanger with spiral baffle plates compared with the conventional heat exchanger is not so large as expected because only one tube and four tube passes system is assumed for the convenience of modeling and computation time. However, it is expected that heat transfer will be enhanced more considerably when the number of tubes increases.

4. CONCLUSIONS

Three-dimensional numerical analyses are performed for the shell-and-tube heat exchangers with spiral baffle plates and conventional vertical baffle plates using the commercial thermal-fluid analysis code, CFX4.2. Shell and tube side flow fields, pressure drops, and heat transfer performances are analyzed. The results of the shell-and-tube heat exchangers with spiral baffle plates are compared with those of the conventional shell-and-tube heat exchanger. The results show that

(1) The shell-and-tube heat exchanger with spiral baffle plates improves heat exchanger performance because rotational flow in the shell caused by the spiral baffle plates eliminates the stagnation region that occurs near the contact regions of the shell and vertical baffle plates in the conventional heat exchanger.

(2) Spiral baffle plates introduce vortices in the shell-side flow and enhance heat transfer between shell and tube side fluids, even though they increase pressure drop significantly.

ACKNOWLEDGMENTS

This study was supported by Korea Institute of Science & Technology Evaluation and Planning(KISTEP) and Ministry of Science & Technology, Korean Government through its National Nuclear Technology Program(Project number:M20362000001-03B0500-02310).

REFERENCES

[1] Walker G. *Industrial heat exchangers - a basic guide*. 2nd ed. New York: Hemisphere Publishing Co., 1990.

[2] Incropera FP, DeWitt DP. *Fundamentals of heat and mass transfer*. 4th ed. New York: John Wiley & Sons Inc., 1996.

[3] Lee SH, Nam SC, Ban TG. *Performance of heat transfer and pressure drop in a spirally indented tube*. KSME International Journal 1998;12(5):917-925.

[4] Lee KS, Kim WS. *The effects of design and operating factors on the frost growth and thermal performance of a flat plate fin-tube heat exchanger under the frosting condition*. KSME International Journal 1999;13(12): 973-981.

[5] Sparrow EM, Reifschneider LG. *Effect of interbaffle spacing on heat transfer and pressure drop in a shell-and-tube heat exchanger*. Int. J. of Heat and Mass Transfer 1986;29(11):1617-1628.

[6] Lee SC, Cho JW, Nam SC. *Effect of baffle parameters on heat transfer in shell-and-tube heat exchangers*. Transactions of KSME Part B 1997;21(1):185-194.

[7] Shah RK, Pignotti A. *Influence of a finite number of baffles on shell-and-tube heat exchanger performance*. Heat Transfer Engineering 1997;18(1):82-94.

[8] Li H, Kottke V. *Effect of baffle spacing on pressure drop and local heat transfer in shell-and-tube heat exchanger for staggered tube arrangement*. Int. J. of Heat and Mass Transfer 1998;41(10):1303-1311.

[9] Prithiviraj M, Andrews MJ. *Three dimensional numerical simulation of shell-and-tube heat exchangers, part Ⅰ: foundation and fluid mechanics*. Numerical Heat Transfer Part A 1998;33:799-816.

[10] Prithiviraj M, Andrews MJ. *Three dimensional numerical simulation of shell-and-tube heat exchangers, part Ⅱ: heat transfer*. Numerical Heat Transfer Part A 1998;33:817-828.
[11] AEA Technology. *CFX4.2 manual*. United Kingdom: 1997.
[12] Patankar SV. *Numerical heat transfer and fluid flow*. Washington, DC: Hemisphere Publishing Co., 1980.
[13] Van Doormaal JP, Raithby GD. *Enhancements of the SIMPLE method for predicting incompressible fluid flows*. Numerical Heat Transfer 1984;7:147-163.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

PLANNING OF FUEL AND ENERGY SUPPLY IN LATVIA BY USING MESAP PROGRAMMING MODEL

Sergejs Vostrikovs and Daniels Turlajs*
Riga Technical University
Department of Heat and Power Engineering
Ezermalas str. 6, LV 1014, Riga
Latvia

Antra Kundzina
Datakom, Ltd.
Department of consultation and projects
Latvia

Abstract

In order to develop different energy and fuel production and consumption scenarios and comparisons in Latvia, sufficiently detailed data characterising energy supply system need to be collected and analysed. Since there are large regional differences in Latvia regarding energy and fuel consumption and structure, the data need to be region-specific. Summarised national fuel and energy balance does not give sufficiently representative overview. Thus information were collected and used as baseline data for modelling with MESAP model. The results of modelling process were practised for strategic analysis of energy problems in Latvia. Besides planning departments of ministries, consulting companies, developmental agencies, research institutions and energy enterprises can use it.
Keywords: modelling, prognosis

INTRODUCTION

Analysis of all programmes and prognosis available in Latvia so far show that the issues of fuel and energy are hardly touched upon. In order to develop different energy and fuel production and consumption scenarios in Latvia and to compare them, MESAP programme was used [1].

DESCRIPTION OF MESAP PROGRAMM

The **M**odular **E**nergy **S**ystem **A**nalysis and **P**lanning (MESAP) software has been designed as a decision support system for energy and environmental management on a local, regional or global scale. MESAP consists of a general information system based on relational database theory, which is linked to different energy modeling tools. In order to assist the decision making process in a pragmatic way MESAP supports every phase of the structured analysis procedure (SAP) for energy planning. MESAP offers tools for demand analysis, integrated resource planning, demand – side management and simulation or optimization of supply systems. In addition, MESAP can be used to set up statistical energy and environmental information systems to

* Phone: +371 9176421, Fax: +371 7089749, E-mail: sesi@rtu.lv

produce regular reports such as energy balances and emission inventories.

The main design principles of MESAP are:

- centralized data management with a standardized data interface and information system capabilities;
- flexibility in the time and regional scale;
- availability of different suitable mathematical methodologies for different scopes of analysis;
- user friendliness concerning data entry, consistency checking and report generation;
- decision support for the scenario technique through transparent case management.

MESAP is based on an independent database management system. Database collects and stores all data necessary for the modeling process. Database is one of the basic elements or module layers of MESAP. MESAP system includes several modules that can be used for energy planning and analysis. Figure 1 shows the architecture of MESAP. With the help of MESAP module **PlaNet**, using the built-in scenario technology, the possible development of the energy system can be simulated. Modeling of energy development scenarios in Latvia used this module.

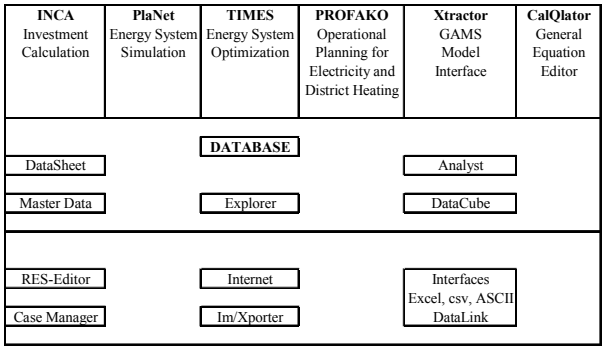


Figure 1: MESAP modeling architecture

Time Series Navigator allows the definition of time series, creation, update, and documentation of data values. **Case Manager** allows creation, description, and registration of scenarios, and to carrying sensitivity analysis. For visualisation of results, creation of tables and graphs, **Analyst** module is used. Input data and the results can be imported and exported from/to MS Excel.

Simulation modules **PlaNet-Flow** and **PlaNet-Cost** calculate energy and emission balance for any energy system, and the necessary capacities for energy transformation technologies. Cost calculation module is able to calculate annual necessary investments, fuel costs, fixed and variable costs for each technology, as well as specific production costs. It allows calculation of all costs of energy system.

FUEL AND ENERGY BALANCE OF LATVIA

Table 1 summarised data on total inland consumption of fuel and electricity.

Energy Resource	Total	Local	Imports
Coal	2.9		2.9
Wood	53.7	53.7	
Peat	1.0	1.0	
Heavy oil	6.2		6.2
Natural gas	60.0		60.0
Electricity	17.6	7.4*	10.2
Other	6.9		6.9
Total	**148.3**	**62.1**	**86.2**

Table 1: Fuel and Electricity Consumption, 2002, PJ [2], [5], [7]

* - Produced in Hydro Power plants

Due to rapid increase in oil product prices, consumption of heavy oil has significantly decreased over the last few years, and utilisation of natural gas has grown. Renewable energy resources accounted for 41% in the fuel balance of Latvia of 2002. Share of wood fuel is increasing. Utilisation of coal and peat is gradually falling, but there is increase in consumption of other fuels, including liquefied gas and diesel. These processes are presented in the Figure 2.

In the electricity balance of Latvia, hydro energy takes the major share. The amount significantly varies depending on the climate. E.g. hydro energy accounted for 47% of the total supply in 1997, 68% in 1998, 43% in 1999, 45% in 2000 and 37% in 2002. It is assumed that the annual average production in hydro power plants is 2200 GWh or 7.4 PJ of electricity.

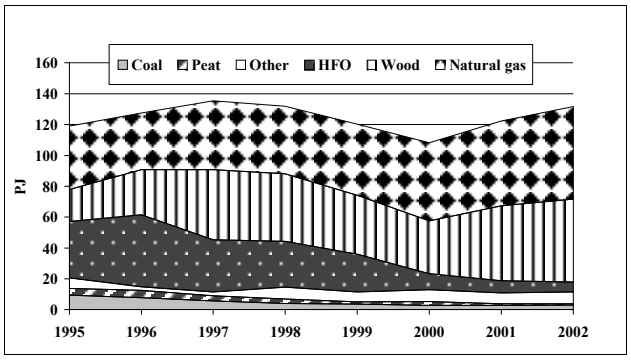


Figure 2: Fuel Balance 1995-2002, PJ

Electricity balance of Latvia in the last few years is presented in Figure 3.

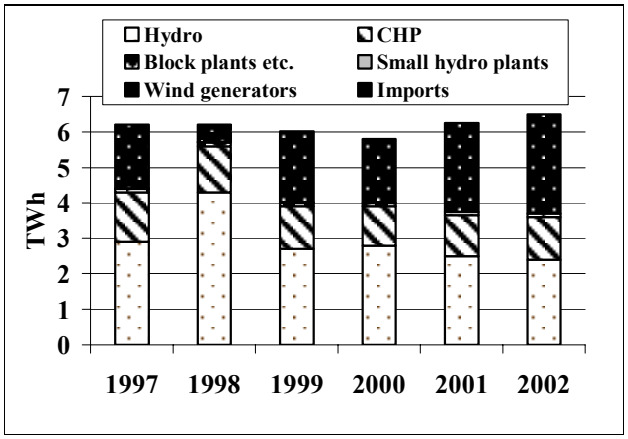


Figure 3: Electricity Balance of Latvia, 1997 - 2002, PJ [5]

ENTRY OF BASELINE DATA IN THE MODEL

Heat Demand

Total heat demand in Latvia increased by 8,5% in 2002, if compared to 2000, and accounted for 7,5 million MWh. Distribution of the demand by regions is presented in the following Table.

Region	2002
Centrs	3691,8
Kurzeme	938,7
Latgale	1444,3
Vidzeme	561,4
Zemgale	552,3
Ziemelvidzeme	311,5
Total	**7500,0**

Table 2: Heat Demand By Regions, Latvia, 2002, GWh

The following distribution of heat demand among consumer groups is used:
- industries 29,5%;
- households 63,6%;
- other consumers 6,9%.

This distribution varies in different regions. The following Table presents distribution of heat demand among consumer groups in different regions.

Centrs	
Industries	16
Households	64
Other consumers	20
Kurzeme	
Industries	27
Households	54
Other consumers	19
Vidzeme	
Industries	23
Households	56
Other consumers	22
Latgale	
Industries	45
Households	42
Other consumers	13

Table 3: Distribution of Efficient Heat Consumption among Consumer Groups in 2002, % [6], [7]

Ziemeļvidzeme	
Industries	35
Households	43
Other consumers	22
Zemgale	
Industries	27
Households	43
Other consumers	30

Table 3 (continued): Distribution of Efficient Heat Consumption among Consumer Groups in 2002, % [6], [7]

Heat losses

In 2002, heat losses accounted for 17,8% in Latvia. This amount has been used for all regions.

Fuel types

The following fuel types are used for the production of heat in the district heating systems: natural gas, heavy oil, coal, diesel, peat, and wood.

Heat is produced in CHP plants and boiler plants.

Electricity Demand

In 2002, electricity demand in Latvia was 5 046 million kWh, which is by 7% more than in 2000. Distribution of electricity demand among regions of Latvia is presented in Table 4.

Region	2002
Centrs	2325,3
Kurzeme	899,6
Latgale	653,2
Vidzeme	436,1
Zemgale	394,8
Ziemeļvidzeme	337
Total	**5046**

Table 4: Electricity Demand by Regions, GWh

Table 5 shows electricity production and supply sources, and the produced electricity in 2002.

Electricity losses

In 2002, electricity losses in Latvia accounted for 16.7%[1]. This amount has been used in calculations for all regions.

Total supply	6,5
Electricity production	3,7
Latvenergo	3,6
Hydro plants	2,4
CHP	1,2
Block plants etc.	0,1
Wind generators	0,002
Imports	2,8

Table 5: Electricity Production and Supply Sources in Latvia, 2002, TWh

End Consumption of Fuel

Total end consumption of fuel was 19.1 million MWh in Latvia in 2002. Of these, 6.6 million were consumed by industries, 8.6 million by households, and 3.8 million by other consumers. The following Table presents distribution of end consumption among regions.

[1] - Latvia has so high electricity losses – the electrical system was built in soviet time and in this time were other norms of efficiency

Region	Total	Industries	House-holds	Others
Centrs	7858	2969	3647	1242
Kurzeme	2475	744	1265	467
Latgale	3047	1370	1253	424
Vidzeme	2501	750	1066	686
Zemgale	1494	372	755	366
Ziemel-vidzeme	1690	442	581	667
Total	**19 065**	**6647**	**8566**	**3852**

Table 6: End Consumption of Fuel in 2002, GWh

DESCRIPTION OF ENERGY PRODUCTION SOURCES OF LATVIA

In the calculation model, the following energy producing units are separated as specific processes: CHP plants *TEC-1*, and *TEC-2*, small CHP plants, hydro power plants of the Daugava cascade, small hydro power plants, wind power plants, and boiler plants grouped according to fuel type. If several fuel types are used, the boiler plant is theoretically divided into parts. Further in the text, each of these processes is described in more detail including the main indicators necessary as input data for the model.

TEC-1 and TEC-2

TEC-1 has four turbine sets, six steam boilers, and two water boilers. Installs electric capacity is 129,5 MW, and heat capacity - 616 MW.
TEC-2 is the largest CHP plant in Latvia. It has four turbines and four water boilers. Installed electric capacity is 390 MW, and heat capacity - 1237 MW.

Item	Unit	TEC-1	TEC-2
Electricity output	GWh	240	923
Heat output	GWh	921	1825
supplied electricity/ supplied heat	GWh/ GWh	0.26	0.48
Total efficiency	%	72	80

Table 7: Main Operation Indicators of TEC-1 and TEC-2, 2002

Small CHP plants

Besides the abovementioned *TEC-1* and *TEC-2* of the electricity supply company *LATVENERGO*,

there are several other CHP plants belonging to industries and heat supply companies.
In 2002, these plants produced about 17 GWh of electricity.
The following Table summarises data on fuel types in these CHP plants, specified by regions.

Region	Natural gas	Heavy oil	Peat
Latgale	80,5	7.2	12.3
Vidzeme	1.0	99.0	0.0
Kurzeme	19.4	80.6	0.0
Zemgale	83.0	17.0	0.0

Table 8: Fuel Mix in Small CHP Plants in 2002, Specified by Regions, %

The indicator of supplied electricity/supplied heat in these plants, compared to *TEC-1* and *TEC-2*, is rather low - 0.02[2] on average.

Hydro power plants of the Daugava Cascade

The Daugava Cascade consists of three hydro power plants: *Plavinu*, K*eguma*, and *Rigas*. Their total installed capacity is 1517 MW. In 2002, output of these hydro power plants was about 2434 GWh of electricity[3].

Small hydro power plants and wind power plants

On Latvian rivers in 2002, there were small hydro power plants with total installed capacity of about 30.1 MW. Annual output is about 60.2 GWh.
Latvenergo owns a wind power plant in Ainazi, with total installed capacity 1.3 MW. In 2002, it produced 2.1 GWh of electricity. Capacity of the wind power plant in Ventspils rajons Uzavas pagasts is 1 MW.

Boiler plants

In 2002, 4,2 million MWh was produced in boiler plants of general use, and 0.9 million MWh – in boiler plants belonging to industries. The following Table summarises data on average efficiency of boiler plants using different fuels.

Fuel	Efficiency, %
Coal	0.55
Natural gas	0.9
Diesel	0.9
Heavy oil	0.84
Peat	0.69
Wood	0.65

Table 16: Boiler Plant Efficiencies, 2002

Emission Factors

The model determines amounts of SO_2, CO_2, and NO_x emissions.
In order to find emission factors, the Second National Report of Latvian under UN General Convention on Climate Change was used [3].
On 10 December 1997, Latvia signed the Kyoto protocol, which requires reduction of greenhouse gas (GHG) emissions. According to the protocol, Latvia needs to reduce emissions of greenhouse gases by 8% by the period 2008-2012. CO2 accounts for major part of GHG (about 90% in 1995). One of the CO2 emission sources is combustion of fossil fuel. Heavy oil causes by 37% more SO2 emissions than the natural gas (78:56,9 kg/GJ). Combustion of coal makes 95 kg CO2/GJ. Thus, to foster fulfilment of Kyoto requirements, utilisation of coal and heavy oil should be reduced, replacing these fuels with natural gas and wood, where possible.

PROGNOSIS ON FIXED MODEL PARAMETERS (ELECTRICITY, HEAT, AND END CONSUMPTION)

Electricity Consumption Prognosis

Table 8 shows electricity consumption prognosis. In the period 2000 to 2020, a small and gradual increase is expected. It will mainly be caused by the consumption growth in household and service sectors. Life standard will increase, and consumers will buy more household appliances, and limit their needs less. No significant changes are expected in the sector of industry. Consumption will also increase in other sectors, e.g. commercial sector, transport, street lighting.

[2] - small CHP plants has very low installed electrical load compared with installed heat load

[3] - the output of electricity from HPP is low compared with installed load because the river Daugava has a small fall, HPP were built in soviet time, too

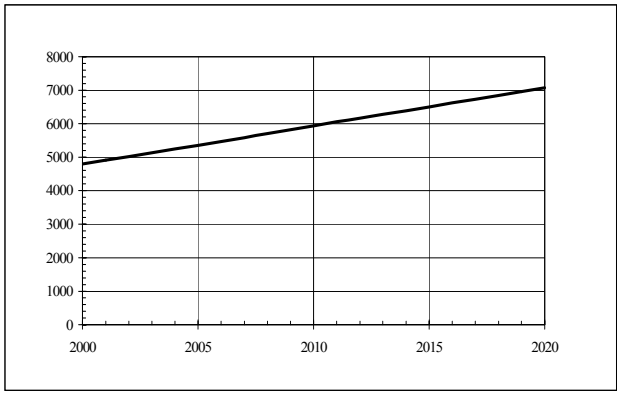


Figure 4: Electricity Consumption Prognosis, GWh

Heat Demand Prognosis

Prognosis say, that the heat demand (Figure 5) could decrease by about 35% in the period 2000 to 2020. There are two basic reasons:
1.decrease of the heat load connected to district heating systems ;
2.efficient heat consumption will be reduced significantly by installation of automatic regulation in heat substations. The process is already ongoing, and it will even speed up due to implementation of heat meters. Consumption of domestic hot water will be the first to decrease, as installation of DHW regulation equipment requires less resource. Efficient heat consumption for heating will fall when independent heating connection will gradually replace current system. Energy efficiency measures in buildings will also contribute to the reduction of heat consumption by reducing the overheating effect.

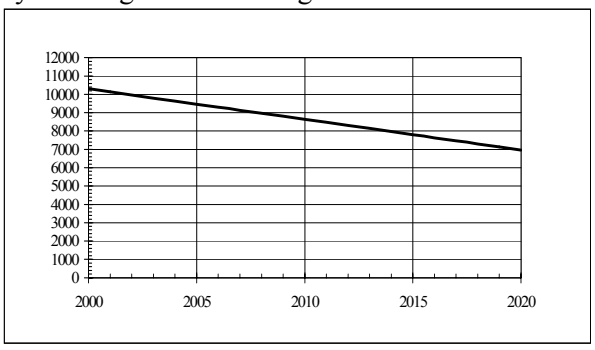


Figure 5: Heat Demand Prognosis, GWh

Fuel End Consumption Prognosis

Current development tendencies show that end consumption of fuel in industry will not change significantly in the modelling period, as neither rapid growth nor decrease in production is expected.

End consumption in households will rise a little due to two factors:

- Increase in living standard. People will not limit their consumption so much;
- Part of the consumers using district heating will disconnect and install their own local heat sources.

End consumption by other consumers will also increase due to:

- New consumers;
- A significant part of consumers of this sector will choose local heat supply.

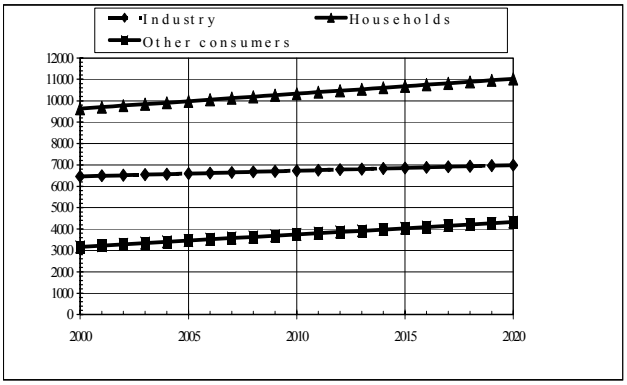


Figure 6: Fuel End Consumption Prognosis, GW

In order to develop different energy and fuel production and consumption scenarios by using a MESAP programming model and comparisons in Latvia, sufficiently detailed data characterising energy supply system need to be collected and analysed.

CONCLUSIONS

The information of energy and fuel production in Latvia were collected and used as baseline data for modelling with MESAP model. The results of modelling process were practised for strategic analysis of energy problems in Latvia. Besides planning departments of ministries, consulting companies, developmental agencies, research institutions and energy enterprises can use it.

REFERENCES

[1] F. Baumhögger, I. Baumhőgger, J. M. Baur, R. Kőhner, U. Schellmann, Ch. Schlenzig, T. Steidle, A. Schweiker MESAP manual // Institute of energy economic and the rational use of energy "IER" University of Stuttgart, 1998

[2] Annual reports on economy development issued by the Ministry of Economy of Latvia, Riga, 2002
[3] Annual national reports of Latvia On Climate Change, Riga, 2000
[4] Annual report of natural gas supplier in Latvia *Latvijas gaze*, 2002
[5] Annual report of electricity supplier *Latvenergo*, 2002
[6] Annual report of Riga district heating company *Rīgas siltums, 2002*
[7] Official energy balance in Latvia, Central Statistics Bureau of Latvia, 2002

Numerical Methods

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

Numerical Simulation of Combustion in Direct Injection Diesel Engines. Application to Additional Heating Systems.

Cédric Garnier*, Jérôme Bellettre, Mohand Tazerout
Ecole des Mines de Nantes
Department of Energetics and Environmental Engineering
La Chantrerie, 4 rue Alfred Kastler BP 20722 44307 Nantes cedex 3
France

Gwénaël Guyonvarch
Valeo Climate Control Inc.
Innovation Department
8, rue Louis Lormand 78321 La Verrière
France

Abstract

This paper focuses on the simulated performance and pollutant emissions of a direct injection diesel engine. A zero-dimensional – two-zone combustion model is described and used to develop a new correlation to predict indicated performance. The combustion model and its hypothesis are presented first. Numerical results are then compared with experiments on a direct injection diesel engine for validation. The model shows that engine performance is well correlated with different running settings. A new correlation to predict the indicated efficiency is thus proposed and its interest is demonstrated. Finally, this study leads to enhance a simulation tool previously created.
Keywords: combustion modelling, simulation tool, cold start, car heating.

Nomenclature

AHS	Additional Heating System
BTDC	Before Top Dead Center
c_p	Specific heat at constant pressure [J/kg/K]
c_v	Specific heat at constant volume [J/kg/K]
d_s	Engine bore [m]
E_a	Activation energy [J/mol/K]
EGR	Exhaust Gas Recirculation
h	Enthalpy per mass unit [J/kg]
h_g	Heat transfer coefficient [W/m^2/K]
k	Reaction rate constant [m^3/mol/s]
LHV	Low Heating Value [J/kg]
m	Mass [kg]
N	Engine rotation speed [rev/min]
P	Pressure [bar]
Q	Heat [J]
R	Gas constant [J/kg/K]
S	Area [m²]
T	Temperature [K]
T_m	Torque [N.m]
u	Internal energy per mass unit [J/kg]
u_{pis}	Mean piston speed [m/s]
V	Volume [m^3]
W	Work [J]
x_b	Burned mass fraction [-]

Greek Letters

Δ	Variation
η_i	Indicated efficiency [-]
η_v	Volumetric efficiency [-]
ϕ	Equivalence ratio [-]
τ_{ID}	Ignition delay [s]
τ_{comp}	Engine compression ratio [-]
σ	Quantity of air at stoechiometry [kg/kg]

* corresponding author : e-mail: cedric.garnier@emn.fr, phone number: (+33) 2.51.85.82.96, fax (+33) 2.51.85.82.99

Introduction

The quality of combustion significantly improved over the last few years thanks to severe antipollution standards (EURO IV in Europe). Consumption benefits are thus results of reduced heat losses towards the coolant circuit and the cabin heating system [1]. During unfavourable atmospheric conditions, acceptable comfort for the user is therefore provided with an Additional Heating System (AHS). Various studies describe cold start thermal deficit and compare the effectiveness of several technologies used by equipment suppliers [1,2,3].

Technical and financial constraints for AHS development have forced the emergence of computer aided engineering. The only constraints remain computing time, flexibility in use and adaptation to different engines. Pirotais [4,5,6] developed a complete simulation tool based on a single zone combustion model linked with nodal method for predicting available heat flux. Its objectives were to simulate the effects of different AHS and analyse global car behaviour during cold starts. The interactions between the engine and cabin heating systems are solved by computing the instantaneous heat transfer from the combustion chamber. The global model architecture (Fig. 1) requires cartography of heat losses towards the coolant loop (Fig. 2), depending on engine load and speed conditions described by a driving cycle. An example of simulation results regarding a warm up test is presented in Fig. 3.

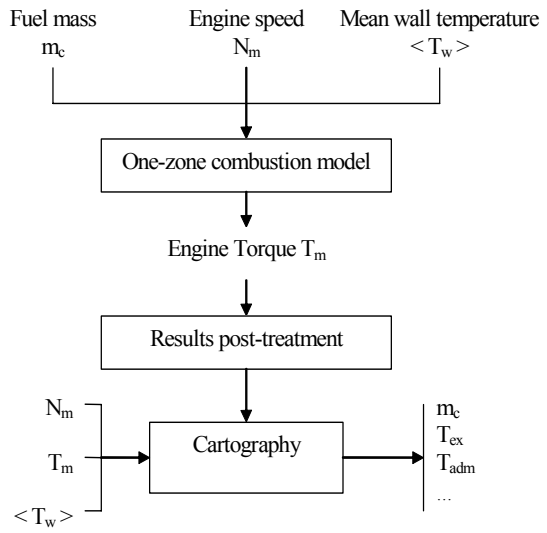


Figure 1 - Global model architecture [6].

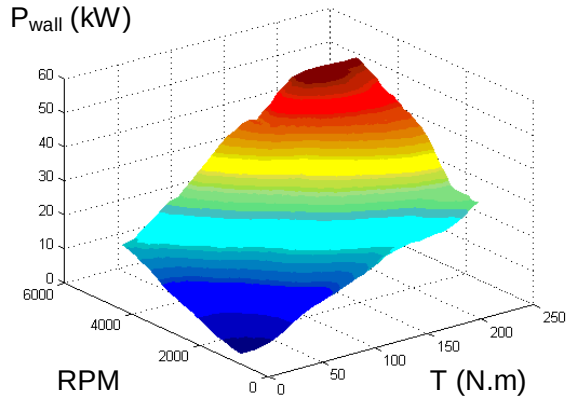


Figure 2 - Cartography of thermal losses towards the coolant circuit [5].

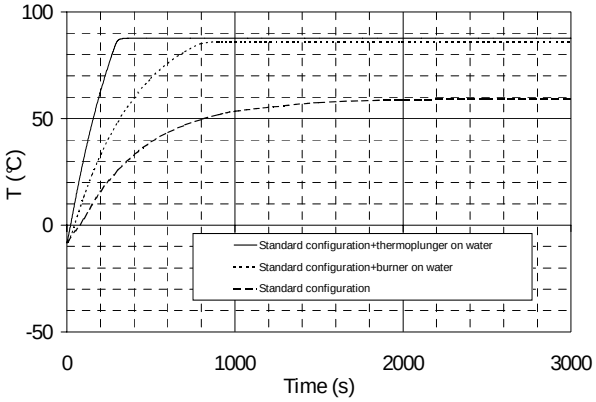


Figure 3 - Predicted impact of additional heating systems [4].

Cartographies are unfortunately unable to take into account variations of other mean parameters (injection timing and profile, mean wall temperature…) without a fastidious battery of simulations.

The objective of this study is to enhance the global simulation tool previously presented [4,5,6]. A "2-zone" combustion model is developed for a greater precision and pollutant formation aspects. Simplified architecture is obtained thanks to a new correlation linking engine settings or parameters, and thermal power lost in the coolant circuit. The final simulation tool architecture is presented in Fig. 4.

The first part of this paper describes the combustion model hypothesis and structure. A description of experimental set-up used to validate the model is then demonstrated. The final part proposes the development of correlations and perspectives.

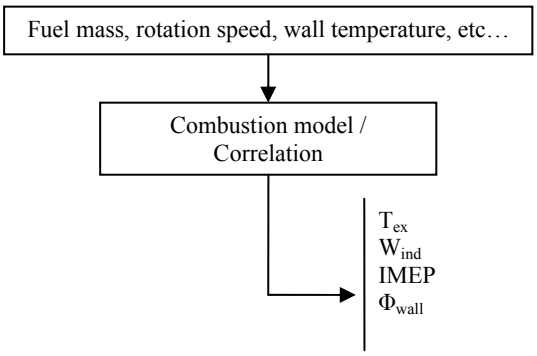


Figure 4 - Final model architecture.

Two-zone model description

General outlines

Two-zone models describe more precisely the combustion process than single ones because chemical reactions are taken into account. The unburned zone is crossed by the flame front where products are formed. They are included in the burned zone, whose temperature T_b is assumed to be homogeneous (Fig. 5). The pressure is assumed to be uniform in the whole chamber.

During admission, compression and exhaust phases, the gas is homogeneous. Its composition is assumed to be constant after combustion (the effect of temperature on molar fractions are not taken into account).

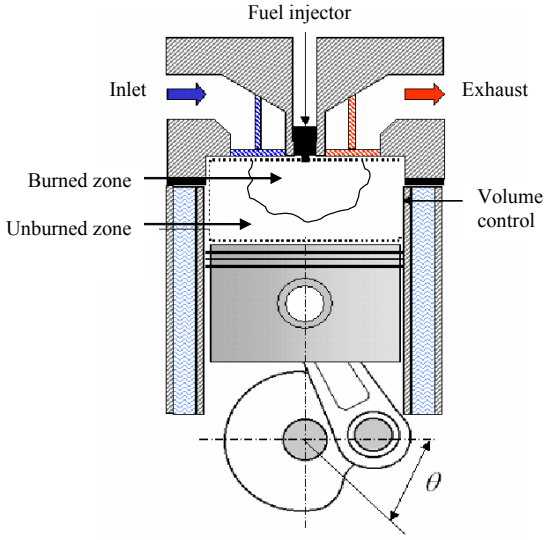


Figure 5 - Two-zone combustion sketch

Mass losses such as blow-by are thus not taken into account in order to simplify equations and reduce computing time. Future studies will be more descriptive on this point.

Equations

The energy conservation applied to each zone gives the differential equations of burned and unburned zones temperature.

$$\frac{dT_u}{d\theta} = \frac{\left(-P\frac{dV_u}{d\theta} - \frac{dQ_{p,u}}{d\theta} - u_u\frac{dm_u}{d\theta} - h_u.m_{cyl}\frac{dx_b}{d\theta}\right)}{m_u.c_{v,u}} \quad (1)$$

$$\frac{dT_b}{d\theta} = \frac{\left(-P\frac{dV_b}{d\theta} - \frac{dQ_{p,b}}{d\theta} - u_b\frac{dm_b}{d\theta} + h_b.m_{cyl}\frac{dx_b}{d\theta}\right)}{m_b.c_{v,b}} \quad (2)$$

The ideal gas law gives the expression of the cylinder pressure.

$$P = \frac{m_u.R_u.T_u + m_b.R_b.T_b}{V_{cyl}} \quad (3)$$

The mass and volume conservation give relations about V_u, V_b, m_u and m_b.

$$V_u + V_b = V_{cyl} \quad (4)$$

$$m_u + m_b = m_{cyl} \quad (5)$$

The burned fraction x_b follows the largely used predictive Wiebe's relation, [8]:

$$x_b = 1 - \exp\left(-a_w.\left(\frac{\theta-\theta_0}{\Delta\theta}\right)^{M_w+1}\right) \quad (6)$$

Parameters of this relation are a_w (fixed at 6,908 for a final burned fraction of 0,999), the crank angle for start of combustion θ_0 and combustion duration $\Delta\theta$. The form factor M_w describes the energy distribution during combustion processes that is Gaussian if equal to unity. In the present case $M_w = 0.9$.

Ignition Delay

Ignition delay is defined as the time between start of injection and start of combustion. Many correlations are proposed in literature. They are validated on different kinds of engines and running conditions. Assanis and al. [9] developed a correlation based on steady-state and transient operations for direct injection diesel engines, that suits to the present study. τ_{ID} depends on the equivalence ratio, the mean temperature *T (K)* and

the pressure *P (bar)* over the ignition interval according to the following equation:

$$\tau_{ID} = 2{,}4.10^{-3}.\phi^{-0,2}.\overline{P}^{-1,02}.\exp\left(\frac{E_a}{R_u.\overline{T}}\right) \quad (7)$$

E_a/R_u is held constant at a value given by Watson (1980) and used by Assanis. Finally combustion starts when condition (8) is reached.

$$\int_{t_{inj}}^{t_{inj}+\tau_{ID}} \frac{dt}{\tau_{ID}(t)} \approx 1 \quad (8)$$

Wall temperature

The wall temperature is naturally non uniform in the whole chamber. However, Pirotais showed that a unique temperature based on a balance carried out on the 3 main areas of the chamber is acceptable [6]. The spatial average considers the piston, the cylinder liner and the cylinder head (with valves). The average time is obtained with integration of the temperature on the entire cycle. The simulation tool then reconsiders this mean wall temperature after each cycle for heat losses calculation (Fig. 6).

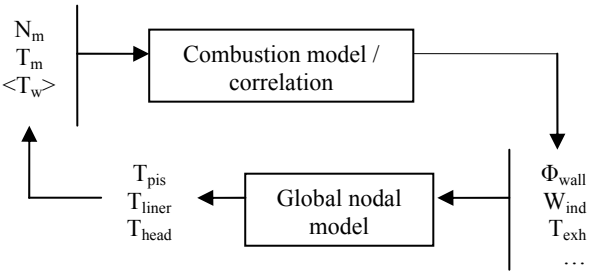


Figure 6 - Computation of mean wall temperature

Heat transfer

The expressions of the wall heat transfer and the convective coefficient are presented in Eq. 9 and 10. The heat transfer between the cylinder trapped mass and the surrounding walls is calculated using the Hohenberg relation [10] (Eq. 10).

$$\frac{dQ_p}{dt} = h_g.S_p.(T_g - <\overline{T}_w>) \quad (9)$$

$$h_g = 130.(10^{-5}.P)^{0,8}.(u_{pis}+1{,}4)^{0,8}.T_g^{-0,4}.V^{-0,06} \quad (10)$$

This correlation is relatively well adapted to direct injection diesel engines since high pressure injection systems allow reduction in soot particles formation. Thus the radiative part of heat transfer is not taken into account in Eq. 10. However, the convective heat transfer will be accurately determined within future studies on current automotive engines. Wall areas of burned and unburned zones are calculated thanks to the burned fraction correlation presented in Eq. 6, [11].

$$S_{p,u} = \left(\frac{\pi.d_s^{\,2}}{2} + \frac{4.V_{cyl}}{d_s}\right).\left(1-\sqrt{x_b}\right) \quad (11)$$

$$S_{p,b} = \left(\frac{\pi.d_s^{\,2}}{2} + \frac{4.V_{cyl}}{d_s}\right).\sqrt{x_b} \quad (12)$$

Chemistry of combustion

The combustion reaction was written for a $C_{10,8}H_{18,7}$ fuel with properties found in [12]. The combustion equation is:

$$\begin{aligned}\phi\varepsilon.C_{10,8}H_{18,7} + (0{,}21.O_2 + 0{,}79.N_2) \rightarrow \nu_1.CO_2 + \\ \nu_2.H_2O + \nu_3.N_2 + \nu_4.O_2 + \nu_5.CO + \\ \nu_6.H_2 + \nu_7.H + \nu_8.O + \nu_9.OH + \\ \nu_{10}.NO\end{aligned} \quad (13)$$

where Φ is the equivalence ratio. CO_2, H_2O, N_2 and O_2 come from the complete combustion, CO, H_2, H, O, OH, and NO from dissociation and recombination reactions. υ_i represents the mole fraction of the constituent i, and ε is the fuel quantity for one mole of products at stoechiometry. Species conservation relations combined with equilibrium constants expressions give a non linear system solvable with different methods. Olikara and Borman [13] described a complete method using a Newton–Raphson algorithm. This method is also used in this study. Specific heat and molar enthalpies expressions are temperature dependent according to relations of Kee and al. [14].

Modelling of nitric oxides formation

In the present study the nitric oxide formation is assumed to follow the largely used extended Zeldovitch mechanism [15]. The following three equations are considered (Eq. 14).

$$N_2 + O \leftrightarrow N + NO$$

$$k_{1,f} = 7{,}6\times10^7.\exp\left(-\frac{38000}{T}\right) \quad (14.a)$$

$$NO + O \leftrightarrow N + O_2$$

$$k_{2,f} = 1{,}5\times10^3.T.\exp\left(-\frac{19500}{T}\right) \quad (14.b)$$

$$NO + H \leftrightarrow N + OH$$

$$k_{3,f} = 2\times10^{8}.\exp\left(-\frac{23650}{T}\right) \qquad (14.c)$$

k_i represents the reaction rate constants. Correlations were taken according to Heywood studies [16].

According to chemical dissociations and assuming that [N] concentration remains constant, the evolution of [NO] concentration is expressed in Eq. 15.

$$\frac{1}{V}\frac{d([NO]V)}{dt} = \frac{2(1-\beta^2)R_1}{\left(1+\beta\frac{R_1}{R_2+R_3}\right)} \qquad (15)$$

where

$$R_1 = k_{1,f}[N_2]_e.[O]_e$$
$$R_2 = k_{2,f}[NO]_e.[O]_e$$
$$R_3 = k_{3,f}[NO]_e.[H]_e$$
$$\beta = [NO]/[NO]_e$$

and index *e* denotes equilibrium.

Time step

A 0,1 crank angle resolution gives a good compromise between computing time and accuracy.

Comparison of calculated and measured data

Test bench

Any prediction model has to be validated thanks to diagrams measured on the test bench. Experimental results on a naturally aspired single-cylinder direct injection diesel engine are available, [17]. The main design parameters are presented in Tab. 1. The complete description of tests can be found in [17].

Constructor	Listter – Petter
Bore	95,2 mm
Stroke	88,94 mm
Displacement	633 cm^3
Compression ratio	19.22 : 1
Cooling system	Forced air circulation

Table 1 – Engine technical features

Comparison

These measurements are compared to the results of the previously described model. An example of pressure diagrams comparison is plotted in Fig. 7.

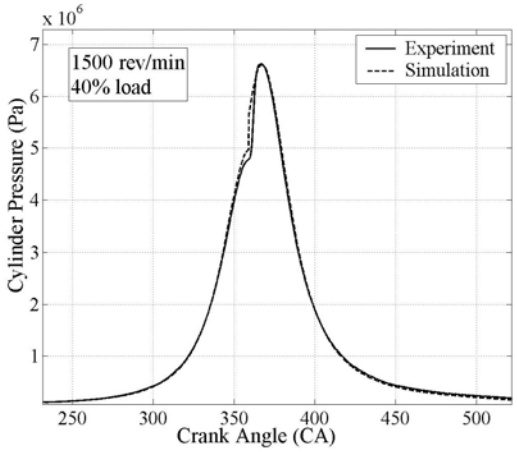


Fig. 7 – Comparison between the experimental and numerical pressure diagrams

Computer modelling gives good agreements with experiments for pressure diagrams and efficiency coefficients. The indicated efficiency η_i, Indicated Mean Effective Pressure *IMEP* and volumetric efficiency η_v comparison are shown in Tab. 2.

	Model	Experiment
η_i *(%)*	50,2	49,7
IMEP (bar)	4,9	5
η_v *(%)*	85,2	85,7

Table 2 – Engine performances comparison (1500 rev/min, 40% load)

In the whole range of tests, the low average error between numerical results and measured values proves reliability of the model.

Simulation results and correlations

Objectives

As already mentioned, the previous simulation program of Pirotais [6] needs a complete cartography of thermal losses towards the coolant loop. The aim of this study is to remove this step by using a correlation for indicated performance, deduced from two-zone modelling results.

Theoretical analysis

IMEP can be expressed from volumetric efficiency, equivalence ratio and indicated efficiency definitions (Eq. 18-22).

$$IMEP = \eta_i . \eta_v . \phi . \frac{LHV}{\sigma} . \frac{P_{atm}}{R_{atm} . T_{atm}} \quad (18)$$

with

$$\eta_v = \frac{m_{air,aspired}}{m_{air,atmospheric\ conditions}} \quad (19)$$

$$\phi = \frac{m_{diesel,injected}}{m_{air,aspired}} . \sigma \quad (20)$$

$$IMEP = \frac{W_{ind}}{V_c} \quad (21)$$

$$\eta_i = \frac{W_{ind}}{m_{diesel,injected} . LHV} \quad (22)$$

Speed and load variations

The experimental design method was applied to determine which parameters have a significant effect on indicated efficiency. The results revealed that speed, load and injection timing have major influence. The effect of rotation speed and load are studied first. The results are presented in Fig. 8.

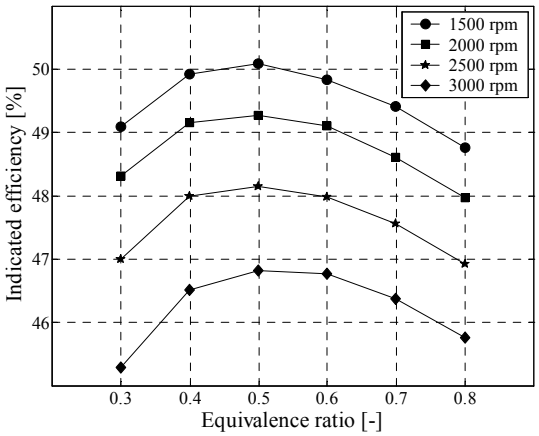


Fig. 8 – Evolution of efficiency with different engine speeds and loads

When the equivalence ratio is under 0.5-0.6, efficiency increases with a negative exponential trend. It decreases for higher values. Considering transient conditions and low loads during cold start driving tests, this study focuses on the first part of the curves.

The relation deduced from results for equivalence ratio under 0.5 is given in Eq. 23.

$$\eta_i = \eta_0 - \alpha . \exp\left(-\frac{\phi}{\phi_0}\right) \quad (\phi < 0.5) \quad (23)$$

Each parameter follows a linear dependence with engine speed (Eq. 24).

$$p_i = A_i + B_i . N \quad (24)$$

where p_i represents η_0, α or Φ_0. A_i and B_i are linear results from fitting database (Tab. 3).

Parameter	A_i	B_i
η_0	53,3	$-2,1.10^{-3}$
α	34,6	$5,3.10^{-3}$
Φ_0	$5,9.10^{-2}$	1.10^{-5}

Table 3 – Linear coefficients for Eq. 24

Considering the whole range of load and rotation speed, the maximum error between simulated and correlated indicated efficiencies remains under 10% (Fig. 9).

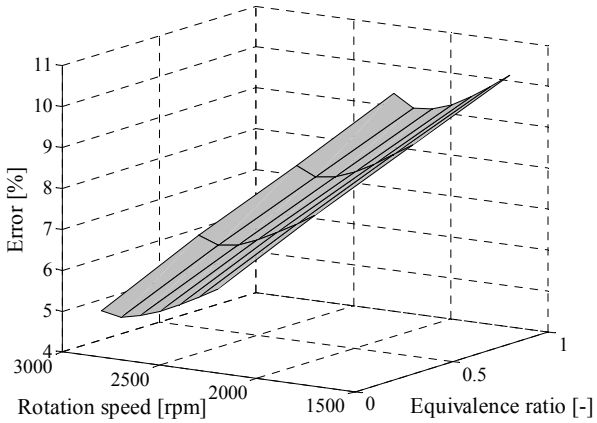


Fig. 9 – Error between simulated and correlated indicated efficiencies

This correlation only takes into account speed and load variations. For combustion engine behaviour, it should be completed by adding other physical parameters. Nevertheless the accuracy of the present correlation proves that this predictive method can be explored.

Injection timing variations

Injection timing and combustion duration are physically linked [16]. However the experimental design method shows that combustion duration has a minor effect on indicated performance compared to injection timing. A sensitivity study on injection timing is then shown.

Fig. 10 presents the evolution of pressure curves with injection timings between 5 and 25 degrees BTDC. As is well known, peak pressure increases with injection timing [7].

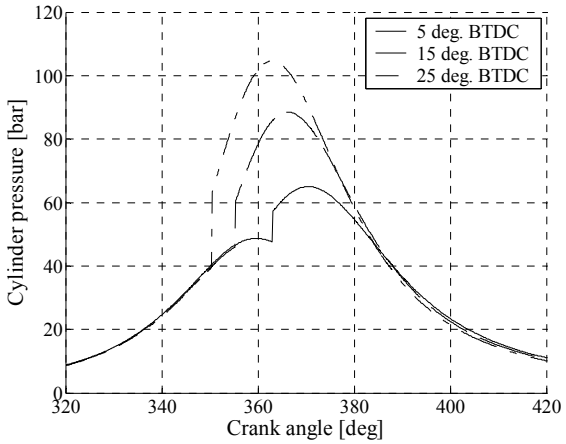


Fig. 10 – Pressure diagrams for different injection timings

Fig. 11 shows the evolution of efficiency with the injection timing in a larger range. Maximum value appears around 15 deg. BTDC. These results show that injection timing must appear in global indicated efficiency correlation. Studies are currently carried out to define how this parameter must be taken into account in the reduced model.

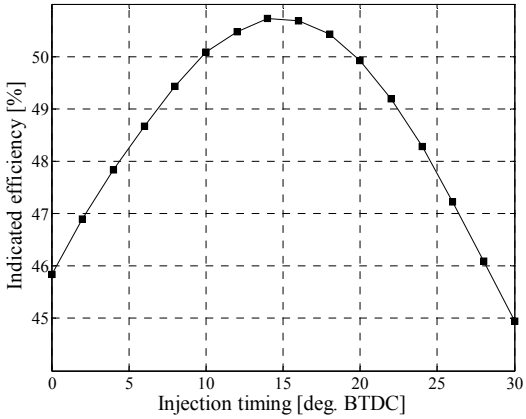


Fig. 11 – Indicated efficiency evolution with injection timing (1500 rpm – 50% load).

Nitric oxide formation

As already shown, the main purpose of this paper is to predict engine performance. Last objectives of this study are AHS development and characterization of their effects on the warm up phase, including emissions. Simulations of NO formation are thus computed. An example is given in Fig. 12. The diagram trend agrees with Rakopoulos results [19] and validates the model. After reduction of the combustion model and its introduction in the global simulation tool, the comparison of the effects on emissions can become decisive for the AHS future classification.

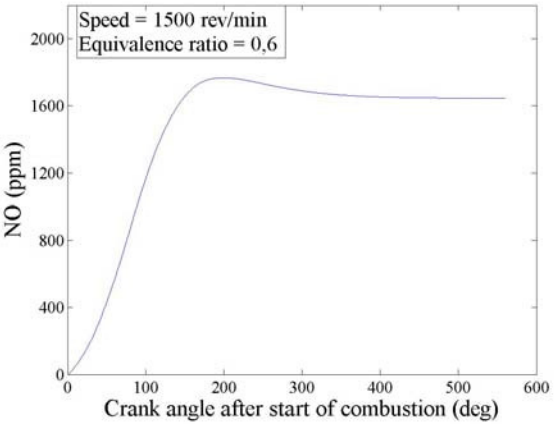


Fig. 12 – Nitric oxide formation (N = 1500 rev/min, φ = 0.5)

Conclusions and perspectives

A correlation for the prediction of indicated efficiency has been demonstrated. This correlation was based on numerical results from a two-zone combustion model detailed in this paper. For the direct injection diesel engine addressed in this paper, simulations proved that performance and engine settings can be correlated. Future studies aim to validate the present method in a larger range of engine types and specially car engines (turbo-charged, direct injection, etc…). Correlations must also include other parameters such as injection timing.

Further studies on the convective transfer coefficient must also be realised. Technical literature proves how difficult it is to well express the wall heat transfer. The most accurate correlation must be adapted and validated on turbocharged direct injection diesel engines. Exhaust Gas Recirculation (EGR) will be taken into account. Its effects on NO, soot emissions and on engine efficiency must appear in the final conclusions of the study. A single fuel injection type was computed here considering the experimental engine equipment. According to real trends several injections can be used for pollutant formation reduction.

References

[1] Haubner FG, Klopfstein S, Koch F. *Cabin heating – a challenge for the TDI cooling system*. Conference SIA 2000/03/23, Lyon, may 2000.

[2] Eilemann A, Kampf H. *Comfort-Management*. SAE Paper N° 2001-02-1738, 2001.

[3] De Pelsemaeker G, Karl S. *Thermal deficit and additional heating systems: actual and future trends*. Conference SIA 2000/03/24, Lyon, may 2000.

[4] Pirotais F, Bellettre J, Le Corre O, Tazerout M, De Pelsemaeker G and Guyonvarch G. *A model of energetic interactions between a car engine, the cabin heating system and the electrical system*. SAE Paper N° 2002-01-2224, 2002.

[5] Pirotais F, Bellettre J, Le Corre O, Tazerout M, De Pelsemaeker G and Guyonvarch G. *A diesel engine thermal transient simulation. Coupling between a combustion model and a thermal model*. SAE Paper N° 2003-01-0224, 2003.

[6] Pirotais F. *Contribution à la modélisation du flux thermique disponible pour le chauffage d'un habitacle d'automobile après un démarrage à froid.* Université de Nantes, PhD Thesis (in french), 2004.

[7] Kouremenos DA, Houtalas DT, Binder KB, Raab A, Schnabel MH. *Using Advanced Injection Timing and EGR to improve DI Diesel Engine Efficiency at Acceptable NO and Soot Levels.* SAE Paper N° 2001-01-0199, 2001.

[8] Rousseau S. *Contribution à l'étude du cycle thermodynamique de moteurs de cogénération fonctionnant au gaz naturel en mélange pauvre.* Université de Nantes, PhD Thesis (in french), 1999.

[9] Assanis DN, Filipi ZS, Fiveland SB, Syrimis M. *A predictive ignition delay correlation under steady-state and transient operation of a direct injection diesel engine.* Paper n°00-ICE-231, ICE-Vol. 33-2, 1999 Fall Technical Conference ASME, 1999.

[10] Hohenberg GF. *Advanced approaches for heat transfer calculations.* SAE Paper N° 790825, 1979.

[11] Tazerout ML. *Etude des possibilités d'amélioration du rendement à charge partielle des moteurs à allumage commandé.* Université Catholique de Louvain, PhD Thesis (in french), 1991.

[12] Turns SR. *An introduction to combustion. Concepts and applications.*McGraw-Hill International Editions, 1996.

[13] Olikara C, Borman GL. *A computer program for calculating properties of equilibrium combustion products with some applications to I.C. engines.* SAE Paper N° 750468, 1975.

[14] Kee RJ, Rupley FM, Miller JA. *The chemkin thermodynamic data base.* Sandia Report, SAND87-8215B – UC-4, 1992.

[15] Lavoie GA, Heywood JB, Keck JC. *Experimental and theoretical study of nitric oxide formation in internal combustion engines.* Combustion science and technology, 1:313-326, 1970.

[16] Heywood JB. *Internal Combustion Engine Fundamentals.* Number ISBN 0-07-100499-8, McGraw-Hill Editions, 1988.

[17] Bilcan A. *Contribution à l'étude du cycle thermodynamique de moteurs fonctionnant en Dual-Fuel.* Université de Nantes, PhD Thesis (in french), 2003.

[18] Monaghan ML. *Engine Friction – A change in emphasis.* The Instm. Mech. Eng. Tribology lecture, 1987.

[19] Rakopoulos CD, Rakopoulos DC, Giakoumis EG, Kyritsis DC. *Validation and sensitivity analysis of a two-zone diesel engine model for combustion and emissions prediction*. Energy Conversion and Management 45 (2004) 1471-1495, 2004.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

NUMERICAL SIMULATION OF COMBINED SCREW COMPRESSOR-EXPANDER MACHINES FOR USE IN HIGH PRESSURE REFRIGERATION SYSTEMS

Ahmed Kovačević, Nikola Stošić, Ian K. Smith,

Centre for Positive Displacement Compressor Technology, City University, London EC1V OHB, U.K.

Tel +44 20 7040 8780, Fax +44 20 7040 8566

e-mail:a.kovacevic@city.ac.uk web:http://www.city-compessors.co.uk

Abstract

Recent interest in natural refrigerants has created a new impetus for studies of CO_2 as a working fluid in vapour compression systems for refrigeration and air conditioning. Two major drawbacks to its use are the very high pressure differences required across the compressor and the large efficiency losses associated with the throttling process in the refrigeration cycle. It is shown how these disadvantages can be minimised by the use of a screw machine both to compress the gas and use the expansion process to recover power. Both these functions can be performed simultaneously, using only one pair of rotors, in a configuration that. partially balances out the forces induced by the pressure difference and hence, reduces the bearing loads to an acceptable level. A further feature is the use of rotors which seal on both contacting surfaces so that the same profile may be used for the expander and the compressor sections. This enables the rotors performing both these functions to be machined or ground in the same cutting operation and then separated by machining a parting slot in them. Computational Fluid Dynamics and Structural Analysis are used in this paper for investigation of the fluid and solid interaction in such machines.

Key Words: Screw Compressor, Screw Expander, Computational Fluid Dynamics, Mathematical Modelling

Nomenclature

C	- turbulence model constants
f	- body force
i	- unit vector
I	- unit tensor
k	- kinetic energy of turbulence
m	- mass
$\dot{m}$	- source in the pressure correction equation
p	- pressure
P	- production of turbulent kinetic energy
q	- source term
$\dot{Q}$	- source in the energy equation
s	- control volume surface
t	- time
u	- displacement in solid
v	- fluid velocity
V	- volume
x	- spatial coordinate
z	- axial coordinate
Γ	- diffusion coefficient
ε	- dissipation of turbulent kinetic energy
ϕ	- variable
λ	- Lame coefficient
μ	- viscosity
η	- Lame coefficient
ρ	- density
σ	- Prandtl number
Δt	- time step used in the calculation

Indices

add	- added or subtracted
eff	- effective
rec	- receiver
T	- turbulent

Introduction

Nearly 20% of the electricity produced in developed countries is used to drive compressors. The majority of these are required for air-conditioning and refrigeration systems. There are therefore considerable environmental advantages to be gained by improving the efficiency of such systems and the compressors that drive them.

A further requirement for environmental protection is the need to minimise the use of refrigerants which may lead to the breakdown of the ozone layer in the upper atmosphere.

As a result of the latter requirement, in the last few years there has been a growth of interest in the use of natural fluids as refrigerants, in place of the halogenated hydrocarbons now widely used. One of these natural refrigerants is CO_2, which is totally free from ozone breakdown effects. Despite the environmental advantages of CO_2 as a working fluid in vapour compression systems for refrigeration and air conditioning, there are two major drawbacks to its use. These are the very high pressure differences required across the compressor, which may be as high as 60 bar, and the large efficiency losses associated with throttling over such a large pressure drop in the near critical region. To overcome the latter, some form of power recovery from this expansion process is essential if the resultant coefficient of performance (COP) is to be acceptable.

A number of proposals have therefore been made to use various types of positive displacement machine, mainly of the vane type, in such a manner that compression in one part of them is combined with recovery of work from simultaneous expansion in another part. However, how well they operate with high pressure differences across the vanes has not yet been confirmed.

For many years, the authors have been investigating the use of twin screw machines as both expanders and compressors of conventional halocarbon refrigerants. These have many potential advantages over other types of positive displacement machine. Unfortunately, when applied to CO_2 the huge bearing forces associated with the pressure distribution within them have hitherto made them appear to be unsuitable.

In this paper, it is shown how twin screw machines can be designed to carry out both the expansion and compression processes in one pair of rotors in such a manner that the rotor forces created by these two processes can be partially balanced to eliminate the axial forces and reduce the radial bearing forces. The disadvantages of twin screw compressors for such high pressure applications are thereby reduced.

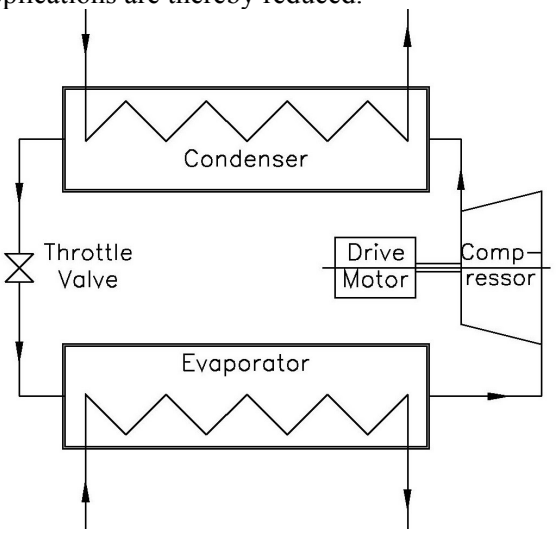


Figure 1 Typical CO2 cycle with throttle valve

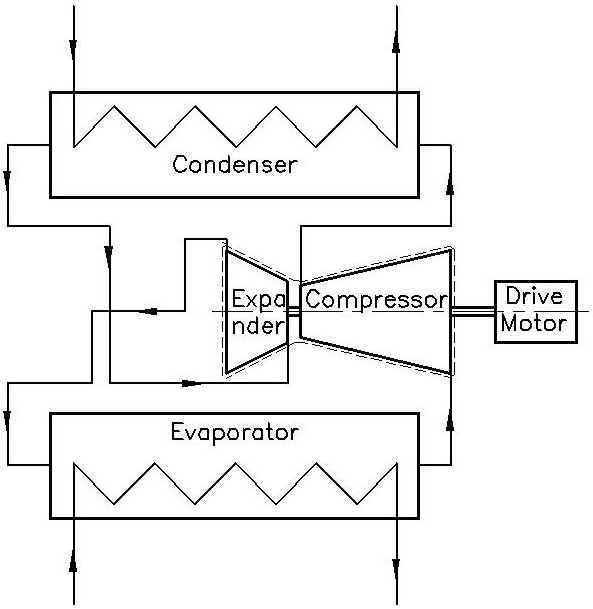


Figure 2 CO2 cycle with combined machine

Analysis of a typical CO_2 system, schematically given in Figure 1, has shown that in an idealised reversible cycle Figure 2, recovery of the throttle losses, by controlled expansion, will increase the COP by as much as 72%. In a practical system, this gain would be reduced by the

compression and expansion efficiencies, which would reduce the expansion work and increase the compression work.

CFD analysis has been used here to model the pressure and temperature changes within the machine. This has been developed further to estimate how the solid components deform as a result of these changes and hence how the fluid-solid interaction resulting from the deformation alters the performance. The results of such numerical simulation are presented in the paper.

Background

Screw compressors are efficient, reliable and compact machines and consequently both the majority of all positive displacement compressors now sold and those currently in operation are of this type. One of the main reasons for their success is the advance in manufacturing techniques, which enable compressor rotors to be manufactured, with very small clearances, at an economic cost. Internal leakages have thereby been reduced to a fraction of their values in earlier designs. However, pressure differences across screw compressor rotors impose heavy loads on them and create rotor deformation, which is of the same order of magnitude as the clearances between the rotors and the casing. Consequently the working pressure differences at which twin screw machines can operate reliably and economically are limited. Current practice is for a maximum discharge pressure of 85 bar and a maximum difference between suction and discharge of 35 bar. *Rinder, [7]*, presented a comprehensive analysis of the effects of such pressures and *Arbon, [1]*, gave a good review of current trends in the design, manufacture and use of high pressure screw compressors. CO_2 (R744) in refrigeration cycles requires both maximum pressures and pressure differences beyond these limits. Accordingly, hitherto screw compressors have not been considered for this purpose.

Other types of positive displacement compressors are used today for compression in CO_2 cycles. Typically, these are single and two-stage reciprocating compressors. A vane compressor study was presented in *[4]*. In such applications the authors concentrate on either the thermodynamic aspects of the CO_2 cycle or mechanical design aspects of its compressors. In considering twin screw machines capable of operation over the pressure range required in CO_2 refrigeration systems, the following factors must be taken into account

Both the screw compressor and the expander consist essentially of a pair of meshing helical lobe rotors contained in a casing, with an open port at one end for admission of the working fluid and at the other for its expulsion. The rotors and housing together form a series of working chambers in which the trapped volume changes from zero to a maximum as the rotors revolve. Thus, the working fluid within them is either compressed or expanded, according to the direction of rotation.

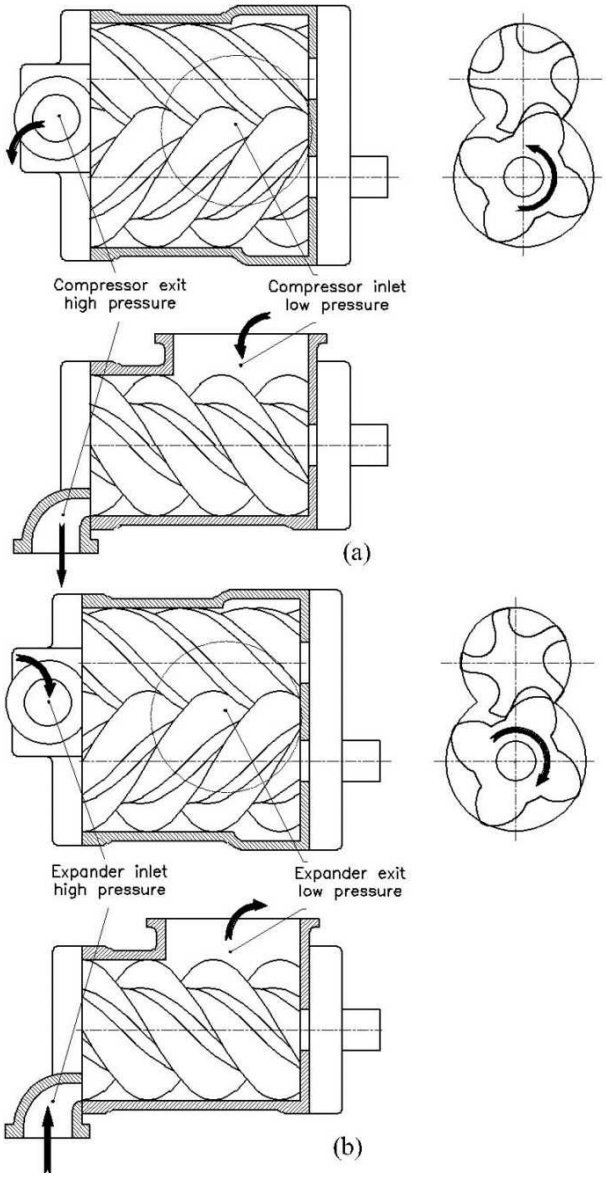


Figure 3 Screw compressor (a) and Screw expander (b) working principles

The compression process is shown in Figure 3a. Here, the suction port is located at the low pressure end, where the trapped volume is largest while the compressed gas is discharged through the high pressure port at a position where the trapped volume is reduced in size. The expulsion of the gas is complete when the trapped volume is zero since the clearance volume in these machines is negligible.

Reversing the direction of rotation, as shown in Figure 3b, causes the direction of fluid flow through the machine to reverse. Gas, then enters through the high pressure port and is discharged through the low pressure port. It then acts as an expander. The machine will also work as an expander when rotating in the same direction as a compressor, if the suction and discharge ports are positioned in the same axial position but on the opposite sides of the casing to those shown, since this is effectively the same as reversing the direction of rotation relative to the ports.

If the rotor profiles are generated to form a seal on both contacting surfaces, then expansion and compression can be performed with equal efficiency in a single pair of rotors rotating in the same direction, simultaneously in one machine, by correct positioning of the appropriate inlet and exit ports along the length of the casing.. There are two methods by which this may be achieved. The first, which is cheaper but more limited in scope, is to carry out both functions in a single chamber, containing only three ports, as shown in Figure 4a. The second is to use separate inlet and exit ports for each function, as shown in Figure 4b, with a division in the casing between the expander and compressor chambers. The common feature of both these arrangements is that the rotors and the casing for both functions can be each machined or ground in a single manufacturing operation.

Both the three port arrangement, as shown in Figure 4(a) and the four port arrangement shown in Figure 4(b) have the advantage that with approximately equal pressures on each end of the rotors, the axial forces on them are more or less balanced out. However, there is an important additional feature of the arrangement shown in Figure 4. This is the location of the high pressure ports, facing each other, in the centre of the casing. By this means the radial loads on the rotors, due to the pressure difference along them, are also partially balanced. This leads to reduced bearing loads and hence, to the ability to design such machines to withstand higher differential pressures than if the expander and compressor were designed and built as separate machines. It also results in smaller mechanical friction losses.

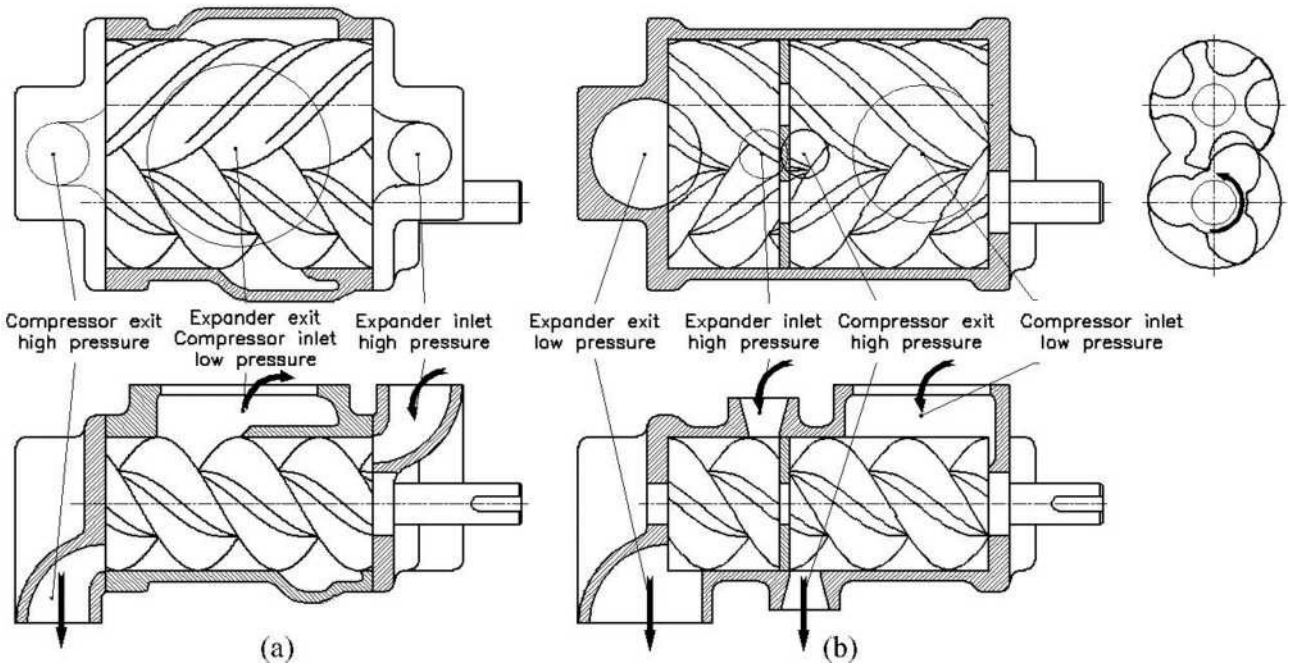


Figure 4 Combined compressor-expander with single chamber (a) and with split chambers (b)

This is of particular importance when considering the design of refrigeration compressors for use with natural refrigerants such as CO_2, where the permitted sustainable bearing load limits the maximum operating pressure difference.

A large mass, but negligible volume, of oil is injected into the casing to seal and lubricate the rotors and also to reduce the temperature rise of the working fluid and hence that of the rotors. Thus thermal distortion of the rotors is small. The net result of reduced axial and radial loading and thermal distortion of the rotors, in the combined machine, is that the range of pressure differences for which it can be used is increased. This makes it possible to consider such a machine for use in CO_2 refrigeration applications.

3-D CCM Calculation

Improvements in both, computational speed and numerical methods over the last thirty years have greatly increased the scope and power of Computational Fluid Dynamics (CFD), and Computational Continuum Mechanics (CCM), in engineering design. Consequently, vendors of CFD and CCM software packages have developed facilities for their use in a wide range of engineering applications. Because none of the standard packages were capable of analysing the complex geometry and processes within screw machines, designers of them have hitherto made little or no use of them.

A CCM method simultaneously enables calculation of the fluid flow and structure behaviour to determine the effects of changes in the compressor geometry on internal heat and fluid flow and vice versa. Such an approach can produce reliable predictions only if calculated over a substantial number of grid points on a sliding and stretching moving mesh.

Grid Generation

The authors have developed an automatic numerical mapping method for arbitrary screw compressor geometry, as explained in *[5]*, which was later used for the analysis of the processes in screw compressors. This method took advantage of the work done by Peric and Demirdzic, *[3]* and *[2]*, who showed that by the use of moving frames on structured and unstructured grids, a common numerical method could be used for the simultaneous solution of fluid flow and structural analysis. On that basis, the authors have developed an interface program called SCORG (Screw COmpressor Rotor Geometry Grid generator), which also enables a grid, generated by the program, to be directly transferred to a commercial CFD or CCM code through its own pre-processor. A number of commercial numerical solvers are currently available, of which the authors decided to employ Star CD's COMET for screw machine calculations. That code offers the possibility to calculate both the fluid flow and its effects on the solid structure simultaneously by the application of CCM.

The interface employs a novel procedure to discretise rotor profiles and to adapt boundary points for each particular application. An analytical transfinite interpolation method with adaptive meshing is used to obtain a fully structured 3-D numerical mesh, which is directly transferable to a CFD code. This is required to overcome problems associated with moving, stretching and sliding rotor domains and to allow robust calculations in domains with significantly different ranges of geometry features.

Governing Equations

The compressor flow and the structure of the compressor parts is fully described by the mass averaged conservation equations of continuity, momentum, energy and space, which are accompanied by turbulence model equations and an equation of state, as shown in [3]. The latter is required to take account of temperature and pressure changes in the screw machines, which alter the working fluid density. In the case of multiphase flow, where oil is injected in the working chamber, a concentration equation is added to the system. The numerical solution of such a system of partial differential equations is then made by inclusion of constitutive relations in the form of Stoke's, Fick's and Fourier's law for the fluid momentum, concentration and energy equations respectively and Hooke's law for the momentum equations of a thermo-elastic solid body.

All these equations are conveniently written in the form of the following generic transport equation:

$$\frac{d}{dt}\int_V \rho\phi dV + \int_S \rho\phi(\mathbf{v}-\mathbf{v}_s)\cdot d\mathbf{s} = \\ = \int_S \Gamma_\phi \operatorname{grad}\phi \cdot d\mathbf{s} + \int_S \mathbf{q}_{\phi S}\cdot d\mathbf{s} + \int_V \mathbf{q}_{\phi V}\cdot dV \quad (1)$$

Here ϕ stands for the transported variable, e.g. Cartesian components of the velocity vector in fluids v_i, enthalpy h, etc. Γ_ϕ is diffusion coefficient. The meaning of source terms, $\mathbf{q}_{\phi S}$ and $\mathbf{q}_{\phi V}$ in all transport equations is given in Table 1.

The resulting system of partial differential equations is discretised by means of a finite volume method in the general Cartesian coordinate frame. This method enhances the conservation of the governing equations while at the same time enables a coupled system of equations for both solid and fluid regions to be solved simultaneously.

Boundary Conditions

This mathematical scheme is accompanied by boundary conditions for both the solid and fluid parts. A novel treatment of the compressor fluid boundaries was introduced in the numerical calculation, as presented in [6]. The compressor was positioned between two relatively small suction and discharge receivers. By this means, the compressor system becomes separated from the surroundings by adiabatic walls only. It communicates with its surroundings through the mass and energy sources or sinks placed in these receivers to maintain constant suction and discharge pressures. The mass source, which exists in the pressure correction equation, is calculated as the difference between the cell pressure p from the previous iteration and the prescribed pressure in the receiver, p_{rec} as:

$$\dot{m}_{add} = \left(\frac{dm}{dt}\right)_{p=const} \approx \frac{p_{rec}-p}{p_{rec}}\,\frac{\rho V}{\Delta t} \quad (2)$$

The volume source for the receiver cells which exists in the energy equation is calculated from the mass source as:

$$\dot{Q}_{add} = h_{add}\left(\frac{dm}{dt}\right)_{p=const} = \dot{m}_{add}\,h_{add} \quad (3)$$

The enthalpy of the fluid added to or subtracted from a cell is calculated on the basis of the thermodynamic property values achieved in the previous iteration.

Table 1 Terms in the generic transport equation (1)

Equation	ϕ	Γ_ϕ	$\mathbf{q}_{\phi S}$	$\mathbf{q}_{\phi V}$
Fluid Momentum	v_i	μ_{eff}	$\left[\mu_{eff}\left(\operatorname{grad}\mathbf{v}\right)^{\mathrm{T}} - \left(\frac{2}{3}\mu_{eff}\operatorname{div}\mathbf{v} + p\right)\mathbf{I}\right]\cdot\mathbf{i}_i$	$f_{\mathrm{b},i}$
Solid Momentum	u_i	η	$\left[\eta\left(\operatorname{grad}\mathbf{u}\right)^{\mathrm{T}} + \left(\lambda\operatorname{div}\mathbf{u} - 3K\alpha\Delta T\right)\mathbf{I}\right]\cdot\mathbf{i}_i$	$f_{\mathrm{b},i}$
Energy	e	$\frac{k}{\partial e/\partial T} + \frac{\mu_t}{\sigma_T}$	$-\frac{k}{\partial e/\partial T}\frac{\partial e}{\partial p}\cdot\operatorname{grad}p$	**T**: grad **v** + h
Concentration	c_i	$\rho D_{i,\text{eff}}$	0	s_{ci}
Space	$\frac{1}{\rho}$	0	0	0
Turbulent kinetic energy	K	$\mu + \frac{\mu_t}{\sigma_k}$	0	$P - \rho\varepsilon$
Dissipation	ε	$\mu + \frac{\mu_t}{\sigma_\varepsilon}$	0	$C_1 P\frac{\varepsilon}{k} - C_2\rho\frac{\varepsilon^2}{k} - C_3\rho\varepsilon\operatorname{div}\mathbf{v}$

The solid part of the system is constrained by both Dirichlet and Neuman boundary conditions through zero displacement in the restraints and zero traction elsewhere. The connections between the solid and fluid parts are explicitly determined if the temperature and displacement from the solid body surface are boundary conditions for the fluid flow and vice versa.

Presentation and Discussion of the Results

A personal computer, with an Athlon 1800 MHz processor and 1.5 GB memory was used for the calculation. The compressor operation was simulated through 24 time steps for one interlobe rotation giving an overall number of 120 time steps for one full rotation of the male rotor. The time step length was synchronised with a compressor speed of 5000 rpm. An error reduction of 4 orders of magnitude was obtained after approximately 50 outer iterations at each time step, which took approximately 10 minutes of computer running time. The overall compressor parameters such as the torque, volume flow, forces, efficiencies and compressor specific power were then calculated. Additionally, pressure-time diagrams of the compression process, the flow and pressure patterns in the compressor chambers and the rotor deformation are provided.

A balanced rotor two-chamber combined compressor expander is considered here for a high pressure CO_2 application. Gears are not used to synchronize the rotors since a fair amount of oil is injected in the system to seal and lubricate the rotors and cool the gas. The oil also maintains the temperature of the rotors at a similar level to that of the main fluid.

As already explained, the two chamber porting arrangement, shown in Figure 4, results in the balancing of the axial load on the compressor rotors, as shown Figure 5. Also, since the radial loads on the compressor and expander sections are located almost opposite each other near the centre of the rotors, the radial loads on the bearings are thereby reduced as shown in Figure 6.

Hence the mechanical friction losses become smaller. The reduction of the bearing load is of particular importance when considering the design of CO_2 since it is the value of the sustainable bearing load which limits the maximum permissible operating pressure difference.

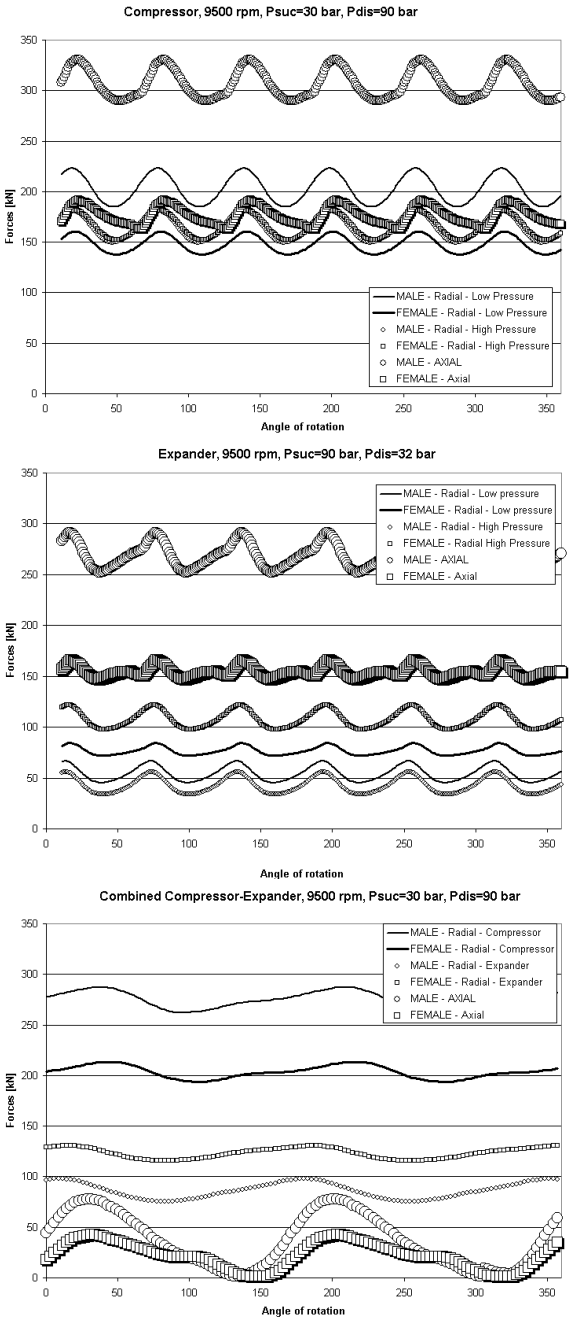


Figure 5 Radial and axial forces for compressor, expander and combined compressor-expander

Additionally, in the combined machine, the deformation of the rotors is substantially smaller than in the case of only compressor or the expander, as can be seen from Figure 7. Deformations presented in this figure are magnified 5000 times. Together, the reduced rotor deformations and the rotor profile used, which results in a stronger female rotor, increase the range of pressure for which a screw machine can be used for CO2 refrigeration applications.

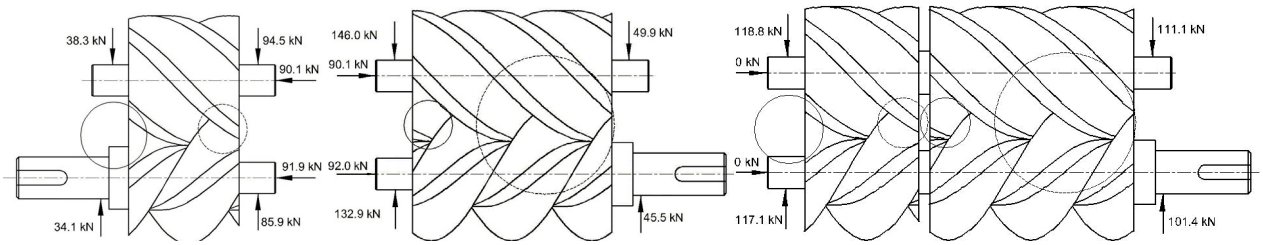


Figure 6 Position of radial and axial forces for compressor, expander and combined compressor-expander

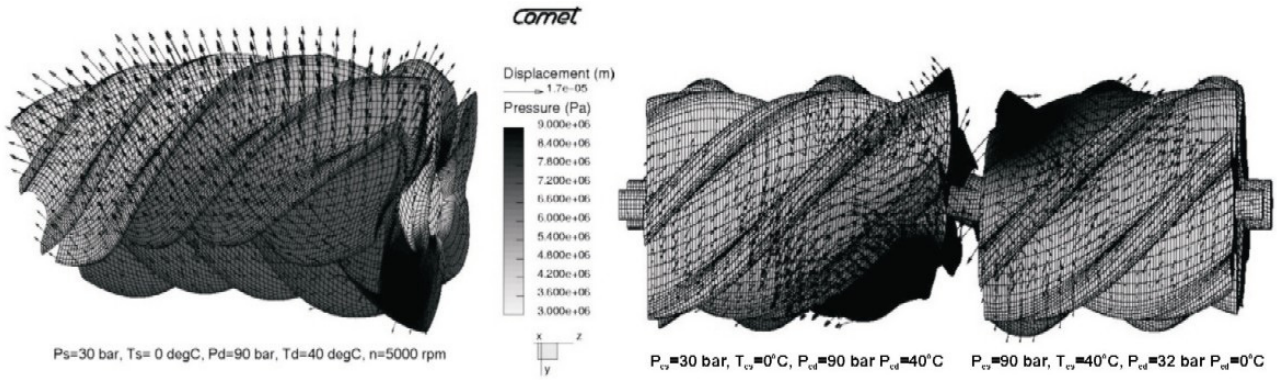


Figure 7 Deformations of the compressor rotors (left) and combined compressor-expander rotors (right)

Conclusions

The ability to use a single pair of screw rotors for simultaneous compression and expansion in separate chambers of the same casing makes it possible to use such machines in vapour compression refrigeration and air conditioning systems with CO2 as the working fluid.

A full 3-D CFD fluid flow and structure simulation has been carried out to determine how pressure and temperature change during the process and how these influence the internal distortion within high pressure screw compressor-expanders. Preliminary results show a reduction in both the radial and the axial forces and thereby a reduction in the mechanical losses. In addition the system efficiency is increased due to partial recovery of the energy in the expander section of the machine.

References

[1] Arbon, I *"High Pressure Screw Compressors"*, IMechE International Conference on Compressors and Their Systems, London, 2001

[2] Demirdzic, I., Muzaferija S., "Numerical Method for Coupled Fluid Flow, Heat Transfer and Stress Analysis Using Unstructured Moving Mesh with Cells of Arbitrary Shape", Comp. Methods Appl. Mech Eng, 1995, Vol. 125, pp 235-255

[3] Ferziger, J.H., Peric, M.; "Computational Methods for Fluid Dynamics", Springer, Berlin, Germany, 1996

[4] Fukuta, M, Yanagisawa, T and Ogi, Y *"Cycle Performance of CO2 Cycle with Vane Compressor-Expander Combination"*, IMechE International Conference on Compressors and Their Systems, London, 2001, 315-324

[5] Kovacevic, A.; Stosic, N.; Smith, I.K.; *"Analysis of Screw Compressor by Means of Three-Dimensional Numerical Modelling"*, International Conference on Compressor and their System, IMechE Conference Transactions 2001-7, London, 2001, p.23-32

[6] Kovacevic A, Stosic N. and Smith, I. K, "*The Influence of Rotor Deflections upon Screw Compressor Performance"*, Conference on Screw Type Machines, VDI-Schraubenma-chinen, Dortmund, Germany, 2002, 17-28

[7] Rinder, L, "*Oil-Injected Screw Compressor of High-Pressure Application",* IMechE International Conference on Compressors and Their Systems, London, 1999, 37-48

[8] Stosic N, Smith, I. K. and Kovacevic A, "*Twin Screw Machines to replace Throttle Valve in Refrigeration Systems"*, Strojniški Vestnik, Journal of Mechanical Engineering, 8 p 484, 2001

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

NEW METHODS FOR NUMERICAL INVESTIGATION OF DYNAMIC PROCESSES
1

Roman Slonevsky
Roksolana Stolyarchuk[1]
National University "Lvivska Politechnika"
Institute of Applied Mathematics and Fundamental Sciences
79013 Lviv, Ukraine

Abstract

The construction of explicit fractional-rational methods for numerical treatment of stiff systems of dynamic processes is discussed. The new formulas of A-stable and L-stable rational approaches are used for one-step (multistep) methods of arbitrary order of accuracy and stability properties are investigated. The results of numerical testing on stiff systems of ODEs are presented.

Nomenclature

A - positively constant matrix
D - domain of existing solution
$D_p(\sigma)$ -operator function
E - identity matrix
f - function of a right-hand part of system
h - step
x - independent variable
$y(x)$ - solution of problem
λ_i - corresponding eigenvalue
Λ - diagonal matrix

Introduction

The mathematical models of dynamic processes in different physical objects have been described by system of ordinary differential equations (ODEs) and formulated as. Modern linear numerical methods for solution Cauchy's problem are share on explicit and implicit methods [1-4]. With opening the stiff phenomena for the systems of ODEs [1] became not effective an explicit numerical methods. In this connection an implicit numerical methods have been widely developed as single-step Runge-Kutta and Rozenbroc – type methods and multistep methods. All these methods based on the repeated solution of nonlinear algebraic equations that is one of the main restrictions for applications of implicit methods. Theoretically these methods under the determinate conditions have posses stability properties, but during their practical realization this properties can be looses. .

The most wide development on today has Gear's implicit multistep methods [3,4]. However, for them Dahlquist's barrier takes place [1], which means that A- stable multistep methods cannot have higher than second order. Except for that the characteristic equations have 'parasitic' roots, which influence on the stability domain of methods.

The new directions in construction of numerical methods for investigation of stiff systems we can find in Wambecq A. [5,6]. He used the Pade transformation for construction nonlinear fractional-rational numerical methods. However, their distributionS were impossible due to operations of division on vector systems of ODEs on vector. But Sottas G. [8] has been shown, that Wambecq's methods are unstable for stiff systems of ODEs.

The resulted brief analysis of the modern numerical methods of solution stiff systems of ODEs has shown the urgency of construction new types and more effective numerical methods.

[1] Corresponding author. E-mail: roksols@yahoo.com

Structure of fractional-rational approaches for stiff problem

We consider system of the ordinary differential equations

$$y' = f(x, y) \quad , \tag{1}$$

where $y \in R^N$, $f : R^{N+1} \to R^N$, on an interval $x \in [0, x_k]$ with initial conditions

$$y(0) = y_0 . \tag{2}$$

We assume existence of continuous and multiple differentiated unique solution of the problem (1),(2) in domain

$$D = \left\{0 \le x \le x_k : \|y\|_D < M_D < \infty\right\}; \|y\|_D = \\ = \max \|y(x)\|_{R^N} \tag{3}$$

We divided an interval of argument change x on some parts $[x_n, x_{n+1}], (n = \overline{0, N_k})$.

Firstly we consider a scalar problem (1), (2) at $N = 1$ and assuming that the following inequalities take place

$$\left|f'(y)\right|_D < M \ , \ f'(y)_D < 0 . \tag{4}$$

We also assume that we know the solution of a problem in grid node x_n. Let construct an approach of the solution $y(x_n + h)$ in vicinity of a point x_n by the help of fractional-rational function

$$y(x_n + h) \approx z_p(x_n + h) = \frac{\sum_{i=0}^{p} b_i h^i}{\sum_{i=0}^{p} c_i h^i}, \tag{5}$$

where b_i , c_i - independent coefficients from $h = x - x_n$.

Coefficients b_i , c_i we shall define from corresponding conditions

$$z_p(x_n) = y(x_n) \ , \ z'_p(x_n) = \\ = y'(x_n) \ , \ldots , \ z^{(p)}{}_p(x_n) = y^{(p)}(x_n) \tag{6}$$

We present the relation (5) as

$$\sum_{i=0}^{p} c_i h^i z_p(x) = \sum_{i=0}^{p} b_i h^i \tag{7}$$

and take p derivatives of (7) on argument x.

Then the k-derivative ($k = \overline{0, p}$) can be written as

$$\sum_{j=0}^{k} C_k^j \Big(\sum_{i=0}^{p} A_i^j c_i h^{i-j}\Big) z_p^{(k-j)}(x) = \sum_{i=k}^{p} A_i^k b_i h^{i-k}, \tag{8}$$

where C_k^j - number of combinations from k elements on j , and $A_i^j = i(i-1)\cdots\big(i-(j-1)\big)$, $C_k^0 = 1$, $A_k^0 = 1$.

Next we write down $p+1$ values of (8) in a point x_n at $h = 0$, taking into account that $A_j^j = j!$

$$\sum_{j=0}^{k} C_k^j \, j! c_j z_p^{(k-j)}(x_n) = k! b_k \tag{9}$$

Using (6) and having entered a designation $y^{(k)}(x_n) = y_n^{(k)}$) we can receive coefficients b_k from (9)

$$b_k = \frac{1}{k!} \sum_{j=0}^{k} C_k^j \, j! c_j y_n^{(k-j)} \ , \text{ or}$$

$$b_k = \sum_{j=0}^{k} \frac{1}{(k-j)!} c_j y_n^{(k-j)} \quad , \quad (k = \overline{0,, p}) \tag{10}$$

The system (10) defines linkage between unknown coefficients b_i and c_i, that is necessary for performance of condition (6).

For obtain approaching of the solution in the grid node x_{n+1} we use the value of fractional-rational function $z_p(x)$ at $h = x_{n+1} - x_n$ and designate it as $y_{n+1}^{[p]}$:

$$y_{n+1}^{[p]} = \Big(\sum_{i=0}^{p} c_i h^i\Big)^{-1} \Big(\sum_{i=o}^{p} b_i h^i\Big) , \tag{11}$$

The formula (11) has $2p+2$ unknown coefficients, overwise the system (10) contains only $p+1$ equations. That's why, for obtain unknown coefficients b_i and c_i it is necessary to accept $p+1$ free parameters in (10). Depending on a choice of separate coefficients for free parameters and their values from (11) and (10) it is possible to receive various formulas of approaches the solutions possessing certain properties of a adjusting and stability.

We now fix as free parameters c_i for accept from (10) coefficients b_i and then after their substitution in (11) we can receive following expression of fractional-rational approaches formed as a result of a choice of free parameters

$$y_{n+1}^{[p]} = \frac{\sum_{j=0}^{p} c_j h^j T_{p-j,n}}{\sum_{i=0}^{p} c_i h^i}, \tag{12}$$

$$T_{k.n} = \sum_{s=0}^{k} \frac{1}{s!} h^s y_n^{(s0)}, \qquad (13)$$

where $T_{p-j,n}$ - is a partial $(p-j)$ Taylor's sum concerning in grid node x_n.

The formula (12) includes all partial Taylor's sums from p to the zero order, where $T_{0,n} = y_n$. Therefore, if we input designation

$$\omega_i = (\sum_{j=0}^{p} cjh^j)^{-1} c_i h^i \quad , \quad (i = \overline{0, p}), \qquad (14)$$

than formula (12) may be rewrite as

$$y_{n+1}^{[p]} = \sum_{j=0}^{p} \omega_j T_{p-j,n} \, . \qquad (15)$$

It is possible to consider it in the certain sense averaging with weight coefficients of Teylor's approaches from zero up to p order of accuracy.

Definition1. The approach (12) is adjusted with p- order if for a difference between the Taylor's formula p-order of accuracy $T_{p,n}$ and fractional-rational approach takes place the following relation

$$\left| T_{p,n} - y_{n+1}^{[p]} \right| = O(h^{p+1}), \, h \to 0 \, . \qquad (16)$$

Theorem 1. If coefficients $c_i (i = 0, p)$ in (12) are satisfy conditions $c_0 = 1$ and $\sum_{i=0}^{p} c_i h^i \neq 0$, $h > 0$ at $h \to 0$ than approach (12) is adjusted with p order of accuracy irrespective of a choice of concrete values of coefficients c_i.

Proof: We shall check the fulfillment of (16)

$$T_{p,n} - y_{n+1}^{[p]} = T_{p,n} - \frac{\sum_{j=0}^{p} c_j h^j T_{p-j,n}}{\sum_{i=0}^{p} c_i h^i} \, . \qquad (17)$$

After simple transformations of (17) we obtain

$$\left| T_{p,n} - y_{n+1}^{[p]} \right| =$$

$$= \frac{h^{p+1} \left| \sum_{j=1}^{p} \frac{1}{(p+1-j)!} c_j y_n^{(p+1-j)} \right| + O(h^{p+2})}{\left| \sum_{i=0}^{p} c_i h^i \right|} =$$

$$= O(h^{p+1})$$

That means the theorem is prooved.

For constructing of solution of system's ODEs we use the structure of relations (11) and (12). In this case the vector-function and Teylor's approaches also are vectors. Therefore coefficients b_i are vectors of dimension N, and parameters c_i - square matrixes of the size $N \times N$, where N- dimension of system (1). In (12) division is considered as product of an inverse matrix of a denominator (it is supposed it nonsingular) on vector - numerator.

In (12) $y_{n+1}^{[p]}$ and $T_{k,n}$ are N- dimension vectors and c_i- square matrixes of dimension $N \times N$. Further we shall accept, that $c_0 = E$, where E- an identity matrix of dimension $N \times N$.

Stability of fractional-rational approaches

The development of numerical methods of the solution of stiff systems has included the stability investigations [8-13]. We consider the selection of values of matrix coefficients c_i for maintenance of required type of stability approaching.

It is known [14], that in vicinity of the solution $y(x, y_0)$, the system (1) is well approximated by linearized system with Jacobi matrix of the right-hand-side of system. Therefore, Jacobi matrix is an important characteristic of system's behavior. From these reasons, matrix coefficients c_i we set as

$$c_0 = E \quad , \quad c_i = \alpha_i J_n^i \, , \quad (i = \overline{1, p}) \, , \qquad (19)$$

where $J_n^i = \left(\left. \frac{\partial f}{\partial y} \right|_n \right)^i$, α_i - scalar parameter. In this case the approach (12) we rewrite as

$$y_{n+1}^{[p]} = \frac{T_{p,n} + \sum_{i=1}^{p} \alpha_i h^i J_n^i T_{p-i,n}}{E + \sum_{i=1}^{p} \alpha_i h^i J_n^i} \qquad (20)$$

It is necessary to choose the values of parameters $\alpha_i \, (i = \overline{1, p})$ for maintenance the stability of approaches (20).

The stability we shall investigate on model system

$$Y' + AY = 0 \qquad (21)$$

where A- positively determined constant matrix.

An approach (20), which is the solutions of system (21) after simple transformations taken form

$$y_{n+1}^{[p]} = \frac{\sum_{j=0}^{p}(-1)^j(\sum_{i=0}^{j}\alpha_i\frac{1}{(j-i)!})h^jA^j}{\sum_{j=o}^{p}(-1)^j\alpha_jh^jA^j}y_n \quad , \qquad (22)$$

where $\alpha_0 = E$.

By using (22) we will estimated the stability of approach (20).

Let for matrix A exists such matrix B, that $B^{-1}AB = \Lambda$, where Λ - is a diagonal matrix. Then by substitution

$$Y = BZ \qquad (23)$$

we obtain N independent the equations

$$z_i' = -\lambda_i z_i \quad , \quad i = \overline{1,N} ,$$

where λ_i - i-eigenvalue of a matrix A. Every eigenvalue we shall consider as complex number.

Similar substituting (23) into (22) we obtain N independent relations of scalar type

$$z_{k,n+1} = \frac{\sum_{j=0}^{p}(-1)^j(\sum_{i=0}^{j}\frac{\alpha_i}{(j-i)!})h^j\lambda_k^j}{\sum_{j=o}^{p}(-1)^j\alpha_j\lambda_k^jh^j}z_{k,n} \quad , \qquad (24).$$

$k = \overline{1,N}$

Thus, stability of approach (20) can be investigated by the scalar equation

$$y' = -\lambda y \qquad (25)$$

with complex value of parameter λ .

Taking into account (22), we can write down an approach of the solution (25) as

$$y_{n+1}^{[p]} = D_p(\sigma)y_n \quad , (26)$$

$$D_p(\sigma) = \frac{\sum_{j=0}^{p}(-1)^j(\sum_{i=0}^{j}\frac{\alpha_i}{(j-i)!})\sigma^j}{\sum_{j=o}^{p}(-1)^j\alpha_j\sigma^j}, \qquad (27)$$

where $\sigma = h\lambda$, $\alpha_0 = 1$.

The stability domain of method is completely define by the operator function $D_p(\sigma)$.

We enter the designation

$$\beta_j = (-1)^j\sum_{i=0}^{j}\frac{1}{(j-i)!}\alpha_i \ , \ (j = \overline{1,p}) \ , \qquad (28)$$

and rewrite (27) as

$$D_p(\sigma) = \frac{1+\sum_{j=1}^{p}\beta_j\sigma^j}{1+\sum_{i=0}^{p}(-1)^i\alpha_i\sigma^i}. \qquad (29)$$

For example, at $\beta_j = 0$, $(j = \overline{1,p})$ we obtain linear system of the algebraic equations from (28) with parameters $\alpha_i : \alpha_o = 1, \alpha_i = \frac{(-1)^i}{i!}, (i = \overline{1,p})$. (30)

At these values of parameters of approach (20) we will obtain

$$Y_{n+1}^{[p]} = (E + \sum_{j=1}^{p}\frac{(-1)^j}{j!}h^jI_n^j)^{-1}(T_{p,n} + \\ + \sum_{j=1}^{p}\frac{(-1)^j}{j!}h^jJ_n^jT_{p-j,n}) \qquad (31)$$

The approaches (31) of higher than 2-nd order of adjusting are $A(\alpha)$ stable and they are expedient on stiff systems with nonoscillating solutions.

For construction approaches of oscillating solutions it is necessary to use A-stable formulas.

For operator function $D_p(\sigma)$ of A-stable approaches in complex plane $\mathrm{Re}(\sigma)$, $\mathrm{Im}(\sigma)$ must satisfy a condition: $|D(\sigma)| < 1$ on half-plane $\mathrm{Re}(\sigma) > 0$ and $|D(\sigma)| = 1$ on an imaginary axis.

It is easy to prove, that operator function of such approaches can be presented as

$$D_p(\sigma) = \frac{Q_p(-\sigma)}{Q_p(\sigma)}, \qquad (32)$$

where $Q_p(\sigma) = 1 + \sum_{k=1}^{p}a_k\sigma^k$, a_k - positive numbers.

For reception from (28) operator function of the specified structure (32) it is necessary to satisfy the following conditions

$$\beta_j = \alpha_j \quad , \quad \alpha_{2k} > 0 \, , \, \alpha_{2k-1} < 0 \, . \qquad (33)$$

Using (28) and (33), we can received system of the linear equations, after solving which we shall determine values of parameters α_i for construction A-stable approaching (20) . Note, that it is possible to build the A-stable fractional-rational approaches of various orders of adjusting with identical denominators.

For example, approaches of 2-4 order of adjusting we can represent as

$$y_{n+1}^{[2]} = \frac{T_{2,n} - \frac{1}{2}hJ_nT_{1,n} + \frac{1}{12}h^2J_n^2T_{0,n}}{E - \frac{1}{2}hJ_n + \frac{1}{12}h^2J_n^2}, \quad (34)$$

$$y_{n+1}^{[3]} = \frac{T_{3,n} - \frac{1}{2}hJ_nT_{2,n} + \frac{1}{12}h^2J_n^2T_{1,n}}{E - \frac{1}{2}hJ_n + \frac{1}{12}h^2J_n^2}, \quad (35)$$

$$y_{n+1}^{[4]} = \frac{T_{4,n} - \frac{1}{2}hJ_nT_{3,n} + \frac{1}{12}h^2J_n^2T_{2,n}}{E - \frac{1}{2}hJ_n + \frac{1}{12}h^2J_n^2}, \quad (36)$$

which has the common operator function

$$D(\sigma) = \frac{1 - \frac{1}{2}\sigma + \frac{1}{12}\sigma^2}{1 + \frac{1}{2}\sigma + \frac{1}{12}\sigma^2}. \quad (37)$$

Similarly we can to construct the A-stable approaches of higher orders.

Except of A-stable approaches, the important group is consist the L-stable approaches. We shall notice, if an operator function of method has coincide with elements of one of two main under-diagonals of Pade's table of approximation exhibitors [1] that satisfy L-stable methods. It is possible to build L-stable approaches.

For example, L-stable approaches of 2-3 nd orders are present as

$$y_{n+1}^{[2]} = \frac{T_{2,n} - \frac{2}{3}hJ_nT_{1,n} + \frac{1}{6}h^2J_n^2T_{0,n}}{E - \frac{2}{3}hJ_n + \frac{1}{6}h^2J_n^2}, \quad (38)$$

$$y_{n+1}^{[3]} = \frac{T_{3,n} - \frac{2}{3}hJ_nT_{2,n} + \frac{1}{6}h^2J_n^2T_{1,n}}{E - \frac{2}{3}hJ_n + \frac{1}{6}h^2J_n^2}, \quad (39)$$

with common operator function

$$D(\sigma) = \frac{1 - \frac{1}{3}\sigma}{1 + \frac{2}{3}\sigma + \frac{1}{6}\sigma^2}. \quad (40)$$

They are especially suitable for research stiffly-oscillating problem.

Similarly, it is possible to build and other formulas of higher orders approaches, possessing set types of stability. For realization of concrete approaches formulas, it is necessary to specify a way of calculation of partial Teylor's sums $T_{k,n}$ and values of Jacobi matrix in the net units.

If for calculation of the specified sums we take an explicit Runga-Kutta methods and formulas of fractional-rational approaches then it is possible to build single-step fractional-rational numerical methods. If for approximation of the partial Taylor's sums we use explicit linear multistep numerical methods, we can construct the multistep fractional-rational numerical methods.

Test Problem

We compute the following nonlinear system of the equations :

$$\begin{aligned} y'_1 &= (b-3a)y_1 - (3a+b)y_2 + (b-a)y_1^2 + \\ &+ (3a+b)y_1y_2 + 4a \\ y'_2 &= (a-b)y_1 + (b-7a)y_2 + (3a-b)y_2^2 + \\ &+ (b-a)y_1y_2 + 4a \end{aligned} \quad (41)$$

Initial conditions $y_1(0) = 1$, $y_2(0) = 1.5$ on an interval of argument change $0 \le x \le 20$ and condition for $a < 0$, $b < 0$.

The system of the equations (41) has an exact solution

$$\begin{aligned} y_1 &= \frac{4 + e^{2ax} - e^{2bx}}{2 + e^{2ax} + e^{2bx}}, \\ y_2 &= \frac{4 + e^{2ax} + e^{2bx}}{2 + e^{2ax} + e^{2bx}}. \end{aligned} \quad (42)$$

In an index point x=0 eigenvalues λ_1 and λ_2 of the Jacobi matrix for system (41) has accept values

$$\lambda_{1,2} = \frac{a + b \pm \sqrt{(a+b)^2 + 8(a-b)^2}}{4} \quad (43)$$

During the problem's solution we yields the limiting values $\lambda_1 \to 2a$ and $\lambda_2 \to 2b$.

The system (41) has been investigated numerically by proposed rational methods and implemented in BASIC computer codes (Program "RROSM") with different accuracy E-02, E-04, E-06 at values $a = -50000$ and $b = -0.5$. In this case the

eigenlavues at x=0 are $\lambda_1 = -49999.832$, $\lambda_2 = 24999.582$ and then aspire to limiting values $\lambda_1 = -100000$ and $\lambda_2 = -1$.

All received results do not exceed the set allowable error on all an interval of integration.

Experimental researches of other various stiff test problems also have confirmed efficiency of fractional-rational suggested methods.

Conclusions

As the results of paper we obtained:

- Developed theory fundamentals of stable fractional-rational approaches for numerical methods of the stiff systems ODEs;
- The type of approaches depends on coefficients of Taylor's sums. Their approximation technique based on one-step and multistep linear numerical methods;
- Realization of proposed methodology for solving the stiff problem convincible implemented in computer code and tested by several examples.

Hhhh

References

[1] Hairer E.,Wanner G. *Solving Ordinary Differential Equations II: Stiff and Differential Algebraic problems.* Berlin. Springer-Verlag, 1996.

[2] Dekker K.,Verver I., *Stability of Runge-Kutta methods for stiff nonlinear differential equations.* Moscow, Mir. 1988 (in Russian).

[3] Gear C.W. *Numerical solution of ordinary differential equations.* SIAMRev.23,1981. pp.10-24.

[4] Gear C.W. *Numerical initial value problems in ordinary differential equations,* Prentice Hall, 253p.p., 1971.

[5] Enright W.H.,Hall T.E., Lindberg B. *Comparing numerical methods for stiff systems of ODEs.* BIT, 1975, 15. 10-48 .

[6] Wambecq A. *Nonlinear methods in solving ordinary differential equations.* Journal of Computational and Applied Mathematics. 1976, V2, №1-27-33.

[7] Wambecq A. *Rational Runge-Kutta methods for solving system of ordinary differential equations.*/ Computing, 1978, 20. pp.333-342.

[8] Sottas G. *Rational Runge-Kutta methods are not suitable for stiff systems of ODE's.* Journal of Computational and Applied Mathematics , 1984 , 10, pp.169-174.

[9] E. Hairer , Lubich & M.Roche *Error of Runge-Kutta methods for stiff problems studied via differential algebraic eguations,* BIT, 1988. Vol.28, pp.678-700,.

[10] E. Hairer, Lubich & M.Roche *Error of Rosenbrock methods for stiff problems studied via differential algebraic eguations,* BIT, vol.29, pp.77-90,1989,

[11] [11]P.Kaps & A.Ostermann *L(α |)-stable variable order Rosenbrock methods.* To appear in :K. Strehmel,ed.,Proceedings of the Halle Seminar 1989 , Teubner Texte zur Mathematik,1990.

[12] Lie I. *The stability function for multistep collocation methods.* Numer.Math.,vol.57, pp.779-787, 1990.

[13] Wanner G. *Characterization of all A-stable methods of order 2m.-4.* Bit,vol.20,pp.367-374,1980.

[14] Bahvalov N., Dzydkov N., Kobelkov G. *Numerical Methods.* FIZMATLIT, Moscow, 2002 (in Russian).

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

A NUMERICAL INVESTIGATION OF THE EFFECT OF INLET POSITION ON THE TRANSIENT TURBULENT FLOW IN A FUEL OIL STORAGE TANK

Nurten VARDAR*
Yildiz Technical University, Faculty of Mechanical Engineering
Department of Marine Engineering

Abstract

The filling process of a fuel oil storage tank containing a higher temperature fuel oil is numerically analysed. The effect of inlet position on transient turbulent flow in a fuel oil tank is investigated by a two-dimensional $k-\varepsilon$ model. A transient stream function-vorticity formulation is used for predicting streamlines and temperature distributions. Streamlines, temperature distributions and transient average temperature variations are presented for four different inlet port locations and for two different aspect ratios. Finally, the effect of aspect ratios on transient average temperature variations in the tank is discussed.
Keywords: numerical, turbulent flow, high viscosity, fuel oil tank

Nomenclature

B	width of tank [m]
$C_{1,3,20,\mu}$	constants
d	depth from the fuel oil surface [m]
E	dimensionless rate of dissipation of turbulence kinetic energy
g	gravity acceleration [m/s^2]
Gr	Grashof number
h	height of inlet port [m]
H	distance from the inlet port to the fluid surface [m]
k	turbulence kinetic energy [m^2/s^2]
K	dimensionless turbulence kinetic energy
L	height of tank [m]
Pr	Prandtl number
Re	Reynold number
Re*	Reynold number based on eddy viscosity
R_F	flux Richardson number
S	turbulence kinetic energy due to Reynold shear stress
t	time [s]
T	temperature [K]
ΔT	$T_i - T_0$
u, v	horizontal and vertical velocity components [m/s]
U, V	dimensionless horizontal and vertical velocity components
x, y	horizontal and vertical coordinates
X, Y	dimensionless horizontal and vertical coordinates
α	thermal diffusivity [m^2/s]
β	volumetric coefficient of expansion
ε	rate of dissipation of turbulence kinetic energy [s^3/m^2]

* Phone : +90 212 2597070x2312 Fax : +90 212 2582157
E-mail : vardar@yildiz.edu.tr

η	kinematic viscosity [m^2/s]
γ	start-up function
θ	dimensionless temperature
$\sigma_{\varepsilon,k}$	constants
τ	dimensionless time
τ_0	dimensionless startup time
τ^*	dimensionless time
ω	vorticity [1/s]
Ω	dimensionless vorticity
Ψ	dimensionless stream function

Subscripts

0	inflow
i	initial
t	turbulent

Introduction

The low grade heavy fuel oil used in Diesel engines on board ships and in power stations include sludge contents, catfines and other harmful solid particles. There is an increased demand for efficient cleaning in order to achieve reliable and economical operation of Diesel engines. The fuel treatment systems include heaters, filters, separators, pumps and settling and daily service tanks. The main purpose of using of settling tanks from cleaning aspects is to act as a buffer tank, to provide a constant temperature and to settle and drain gross water contaminant.

According to the classification societies a special fuel supply is to be provided for the prime movers of the emergency source of electrical power, for example the motors driving emergency fire pumps. By the arrangement of the fuel oil tank, the emergency diesel equipment must be kept in a state of readiness when the outside temperature is low [1]. On the other hand the temperature in a tank should not be below 50-60 ^oC, otherwise the fuel oil may not be pumpable [2]. Because of that it is important to maintain a constant temperature and a temperature control system should be installed to minimise the temperature fluctuations in a tank. Due to the potential risk of thermal stratification problem in a tank and too low temperature at suction point for a feed pump the position of inlet port of a tank is very important [3].

The stratification of two fluids at different temperature is an important consideration in the design of thermal storage tanks. In recent years considerable research efforts have been devoted to the study of thermal behaviour of the stratified fluids in thermal storage tanks especially in solar energy domain. Gari and Loehrke [4], Baines et al. [5], Abu-Hamdan et al. [6] and Nakos [7] investigated the flow conditions into tanks. Homan and Soo [8, 9] modelled the transient stratified flow into a chilled-water storage tank for laminar flow conditions. Mo an Miyatake [10] numerically analysed the transient turbulent flow field in thermally stratified water tank. Cai and Stewart [11] investigated the mixing process that occurs when a cold fluid flows in to a two dimensional tank containing a warmer fluid. A few studies have been reported for the case of different fluids such as liquid natural gas or oil [12-14]. Vardar [15] numerically analysed the filling process of a fuel oil storage tank with an inlet port close to the bottom of the tank and presented streamlines, isotherms and transient average temperature variations for three different inflow velocities and for three different cases of heating.

The purpose of this present work is to investigate the effect of inlet position and aspect ratios, *L/B*, on the transient turbulent flow field and on the temperature distribution in a fuel oil storage tank.

Mathematical model

A simplified two-dimensional tank geometry is shown in Fig.1. The height and the width of the tank are *L* and *B* respectively. The height of the inlet is *h* and the distance from the inlet port to the fluid surface is *H*.

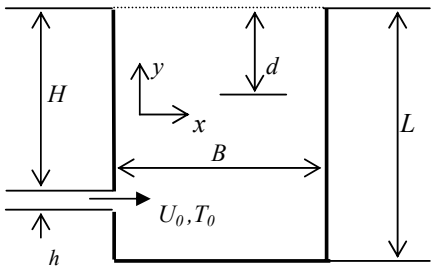


Figure 1: Geometry and coordinates of the tank

It is assumed that the side and bottom walls and fluid surface are thermally insulated and that the tank is filled with hot fuel oil at an initial temperature. Cold fuel oil, which is at a lower temperature than the fuel oil in the tank, enters from the inlet port at the left hand side of the tank.

The governing equations describing transient and turbulent flow of viscous and incompressible fluid in the tank are continuity, momentum, energy, turbulent kinetic energy and dissipation rate of the turbulent kinetic energy equations. A more detailed explanation about the mathematical model can be found in a previous study of the author [15]. Dimensionless equations in stream function-vorticity formulation are given as follows.

$$\frac{\partial\Omega}{\partial\tau}+\frac{\partial(U\Omega)}{\partial X}+\frac{\partial(V\Omega)}{\partial Y} = \frac{\partial^2}{\partial X^2}(\Gamma\Omega)+\frac{\partial^2}{\partial Y^2}(\Gamma\Omega)+\frac{Gr}{\mathrm{Re}^2}\frac{\partial\theta}{\partial X}+S \quad (1)$$

$$\frac{\partial\theta}{\partial\tau}+\frac{\partial(U\theta)}{\partial X}+\frac{\partial(V\theta)}{\partial Y} = \frac{\partial}{\partial X}\left(\Gamma\frac{\partial\theta}{\partial X}\right)+\frac{\partial}{\partial Y}\left(\Gamma\frac{\partial\theta}{\partial Y}\right) \quad (2)$$

$$\frac{\partial K}{\partial\tau}+\frac{\partial(UK)}{\partial X}+\frac{\partial(VK)}{\partial Y} = \frac{\partial}{\partial X}\left(\Gamma\frac{\partial K}{\partial X}\right)+\frac{\partial}{\partial Y}\left(\Gamma\frac{\partial K}{\partial Y}\right)+S+B-E \quad (3)$$

$$\frac{\partial E}{\partial\tau}+\frac{\partial(UE)}{\partial X}+\frac{\partial(VE)}{\partial Y} = \frac{\partial}{\partial X}\left(\Gamma\frac{\partial E}{\partial X}\right)+\frac{\partial}{\partial Y}\left(\Gamma\frac{\partial E}{\partial Y}\right)$$

$$+C_1\frac{E}{K}(S+B)(1+C_3R_F)-C_2\frac{E^2}{K} \quad (4)$$

$$\Delta^2\Psi=-\Omega \quad (5)$$

$$R_F=-\frac{B}{B+S} \quad (6)$$

$$\Gamma=\frac{1}{ab}\frac{1}{\mathrm{Re}\,\mathrm{Re}^*},\quad B=\frac{1}{\mathrm{Re}^*\,\mathrm{Pr}_t}\frac{Gr}{\mathrm{Re}^2}\frac{\partial\theta}{\partial Y} \quad (7)$$

Eqs.	Ω	θ	K	E
a	1	Pr	1	1
b	1	Pr_t	σ_k	σ_ε

Table 1: Definitions for a and b in Eq. (7)

$$S=\frac{1}{\mathrm{Re}^*}\left\{2\left[\left(\frac{\partial U}{\partial X}\right)^2-\left(\frac{\partial V}{\partial Y}\right)^2\right]+\left(\frac{\partial U}{\partial Y}+\frac{\partial V}{\partial X}\right)^2\right\} \quad (8)$$

Eqs. (1)-(8) are written in terms of the following dimensionless variables :

$$X,Y=\frac{x,y}{h},\ U,V=\frac{u,v}{u_0},\ \tau=\frac{u_0t}{h},\ \Omega=\frac{\omega}{u_0}h \quad (9)$$

$$K=\frac{k}{u_0^2},\ E=\frac{h}{u_0^3}\varepsilon,\ \theta=\frac{T-T_0}{\Delta T},\ \mathrm{Pr}=\frac{\eta}{\alpha} \quad (10)$$

$$\mathrm{Pr}_t=0.75+\frac{0.708}{\log\mathrm{Pr}+2.83},\ Gr=\frac{g\beta\Delta Th^3}{\eta^2} \quad (11)$$

$$\mathrm{Re}=\frac{u_0}{\eta}h,\ \mathrm{Re}^*=\frac{u_0}{\eta_t}h=\frac{1}{C_\mu}\frac{E}{K^2} \quad (12)$$

The turbulent viscosity η_t in Eq. (12) is expressed as follows:

$$\eta_t=C_\mu\frac{k^2}{\varepsilon},\ C_\mu=C_{\mu0}\exp\left[\frac{-2.5}{(1+\mathrm{Re}_t/50)}\right] \quad (13)$$

Re^* is the Reynold number based on eddy viscosity [16]. Pr_t is the turbulent Prandtl number [17] , dimensionless factors C_2 and C_μ are functions of turbulent Reynold number, Re_t [18].

$$C_2=C_{20}\left[1.0-0.3\exp-\mathrm{Re}_t^2\right],\ \mathrm{Re}_t=\mathrm{Re}\frac{K^2}{E} \quad (14)$$

The values of the numerical constants appearing in the above equations are given in Table 2.

C_1	C_3	C_{20}	$C_{\mu0}$	σ_k	σ_ε
1.44	0.8	1.92	0.09	1.0	1.3

Table 2 : Values of the numerical constants

Initial and boundary conditions

The initial conditions:

$$U=V=K=E=\Omega=\Psi=0 \text{ and } \theta=1 \quad (15)$$

At the inlet [11]:

$$U=\frac{2(h/h-Y)Y}{(h/h)^2},\quad \Psi=\int_0^{h/h}UdY$$

$$\Omega = -\frac{\partial U}{\partial Y},\ K = 0.01,\ \frac{\partial E}{\partial X} = 0 \qquad (16)$$

At the left hand wall of the tank a start-up function, γ is used for ramping the inlet boundary condition on velocity from its initial condition to its steady-state value over a dimensionless time period of $\tau_0 = 0.2$, which is defined as follows [8,9]:

$$\Psi = \gamma(\tau, \tau_0) \quad (0 \le H \le H)$$
$$\gamma = \begin{cases} \left(\tau/\tau_0^{\,2}\right)^2 (3 - 2\tau/\tau_0) & \tau/\tau_0 < 1 \\ \Psi(0,h) & \tau/\tau_0 \ge 1 \end{cases} \qquad (17)$$

All of the other walls of the tank are assumed to be no-slip, impermeable and adiabatic.

At the fluid surface:

$$U = E = K = \Omega = \frac{\partial V}{\partial Y} = \frac{\partial \theta}{\partial Y} = \frac{\partial \Psi}{\partial Y} = 0 \qquad (18)$$

Numerical method

The mathematical model is solved by a finite differences procedure which is developed for turbulent flow [15]. The procedure is developed by modifying a method which is described for laminar flow by Chow [19]. At any time instant the vorticity, the temperature, the turbulent kinetic energy and dissipation distributions are obtained from conditions at the previous time step. Upwind and central differencing scheme is used to approximate the time derivatives. Stream function is computed based on vorticity distribution by solving Eq. (5) with successive over relaxation method. Velocity components are calculated from the known stream functions.

The two-dimensional computational region as shown in Fig. 1 is an open top rectangular tank and it has a height of 20 times the dimension of the inlet opening. For the simulations 21X41and 41X41 uniform grid sizes are chosen for two different tank sizes (L/B=2 and L/B=1) after performing the grid independency tests. During the transient process grid size effect is evident but not too significant. The maximum deviations between average temperatures in the tank depending on time is less than 1.3 % even for the grid size of 21X41/51X101.

For the validation of the method the results of the present study are compared with the results of the previous studies and they are found to be in good agreement. [11, 15].

Results and discussion

The tank is assumed to be filled with stagnant fuel oil at 60 ^{o}C. The fuel oil entering into the tank is 40 ^{o}C during the process. Kinematic viscosity of the fuel oil is 3.75X10^{-4} m^2/s and Prandtl number is 10400 at inflow temperature. The inlet height is 0.1 m. Because the inflow velocity into the fuel oil tanks should not be higher than 1.0 m/s [3], the inflow velocity is assumed to be 0.5 m/s. Reynold number related to inflow velocity which is calculated according to Eq. (12) is 133. The value of the parameter of Gr/Re^2, which is an important parameter for the transient turbulent flow in thermal storage tanks, is determined to be 0.055 by the Eq. (11).

The dimensionless time step is chosen to be 0.05. In order to examine the effect of the inlet position and aspect ratios on the transient thermal behaviour of the tank, the average temperatures are calculated at three different depths of the tank and at the surface. The average temperature at a horizontal plane at any depth of the tank is defined as follows :

$$\theta_{ave} = \frac{1}{B/h} \int_0^{B/h} \theta dx \qquad (19)$$

In order to investigate the thermal behaviours of the tank the dimensionless transient average temperatures are plotted versus a new dimensionless time:

$$\tau^* = \frac{t}{BL/u_0 h} = \frac{\tau}{BL/h^2} \qquad (20)$$

When this new dimensionless time, τ^*, is equal to 1.0, it means that the fuel oil in the tank is completely replaced by the fuel oil entering the tank [10, 16].

Effect of inlet position on the transient average temperature variations

In order to see the thermal behaviours of the tank the dimensionless average temperatures are plotted versus the dimensionless time. The variations of average temperatures in three depths

of the tank (d/L=0.25, d/L=0.5, d/L=0.75) and on the fluid surface (d/L=1) for the aspect ratio of L/B=2 and for four different inlet positions (H/h=19, 14, 9 and 4) are shown in Figs. 2-5. For each inlet position the average temperatures at each depth decrease from 1.0 (dimensionless hot fluid temperature) to a lower temperature after a lag time that depends on inlet position.

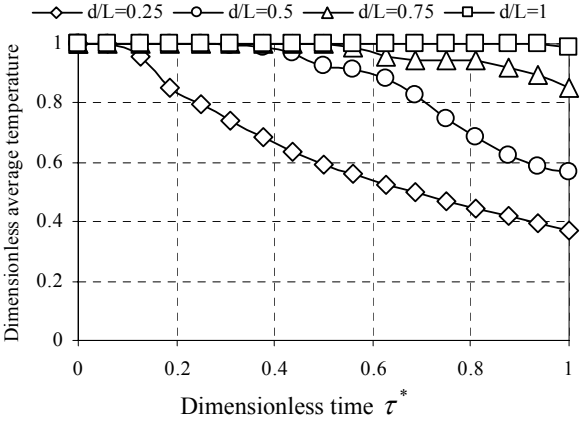


Figure 2 : Transient average temperature variations at four depths for *L/B*=2, *H/h*=19

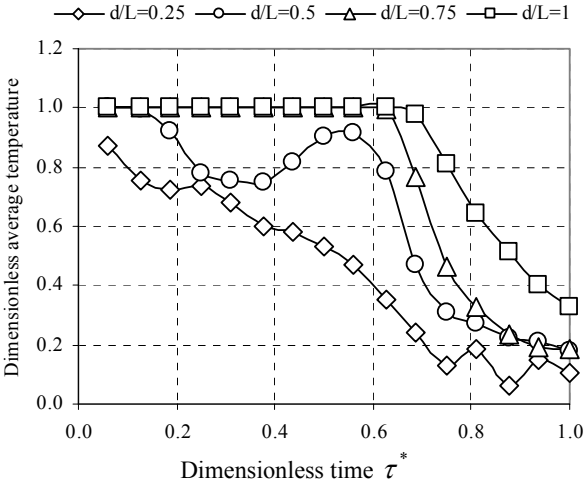


Figure 3 :Transient average temperature variations at four depths for *L/B*=2, *H/h*=14

As shown in Fig. 2, the lag time at *d/L*=0.25 is less than 0.1 and as soon as the lag time ends, the transient average temperature decreases rapidly from 1 to 0.37. The lag times are 0.37 for *d/L*=0.5 and 0.56 for *d/L*=0.75, and the average temperatures for these two depths decrease to 0.56 and to 0.85 respectively at $\tau^* = 1$. The average temperature at the fluid surface stays constant at initial temperature till $\tau^* = 0.95$ and then begins to decrease slightly. As can be seen from Fig. 2 the transient average temperatures at each depth are rather different from each other. As can be seen from Fig. 3 the transient average temperatures at *d/L*=0.75 and *d/L*=1.0 remain constant at the initial temperature till $\tau^* = 0.6$ and then they begin to decrease rapidly to the temperature values of 0.18 and 0.33 respectively. For the case of *d/L*=0.25 the average temperature values almost regularly decrease from 0.87 to 0.11. The average temperature values for the case of *d/L*=0.5 remain constant till τ^*=0.12. After that point on they begin to decrease to a temperature value of 0.18 at τ^*=1.

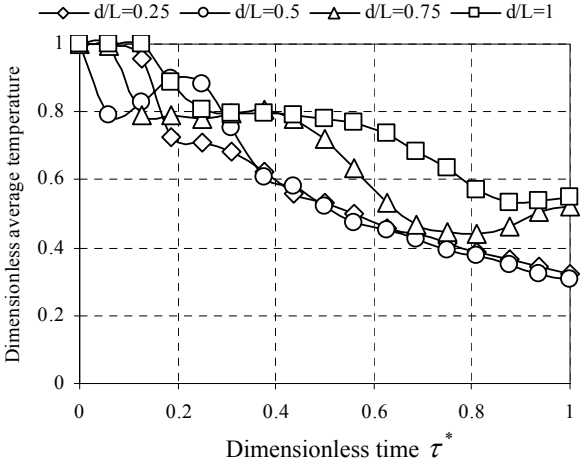


Figure 4 :Transient average temperature variations at four depths for *L/B*=2, *H/h*=9

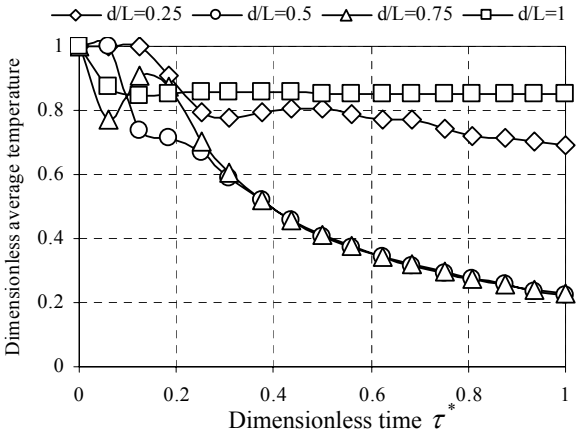


Figure 5 :Transient average temperature variations at four depths for *L/B*=2, *H/h*=4

As can be seen from Fig. 4 the transient average temperatures at each depth of the tank decrease to the values between 0.55 and 0.3 beginning from τ^*=0 and almost a homogeneous temperature distribution occurs in the tank at τ^*=1. As shown in Fig. 5 the average temperatures for *d/L*=1 drop a little at the beginning of the process and remain constant at a value of 0.87 till τ^* =1. The average temperatures for *d/L*=0.25 decrease slightly to a value of 0.69 after the lag time. The average temperatures for the other two depths (*d/L*=0.75 and *d/L*=0.5) are almost the same as each other after τ^*=0.25 and they decrease rapidly to the dimensionless average temperature value of 0.25 at τ^*=1. Therefore the average temperatures at the region close to the middle of the tank are quite different from those at the bottom and top regions of the tank.

Transient flow and temperature fields

For *Re*=133 and Gr/Re^2=0.055 the computed results for different inlet positions are presented as streamlines and temperature fields at τ^*=1 in Figs. 6-9.

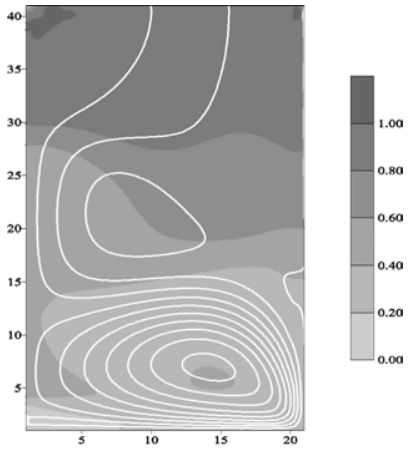


Figure 6 : Streamlines and temperature fields at L/B=2, H/h=19

As can be seen from Fig. 6 cold fuel oil enters into the tank from lower corner of the left-hand wall of the tank, then it impinges upon the right-hand wall and changes its direction to the left-hand wall. After arriving the left-hand wall it returns to right-hand wall and a second vortex forms in the tank. It can be seen that the temperature values decrease in the lower region of the tank and the dimensionless temperature values are more than 0.8 in the upper region of the tank. Above a temperature value of 0.8 the streamlines indicate that the flow is nearly uniform while the region below is filled with recirculation cells.

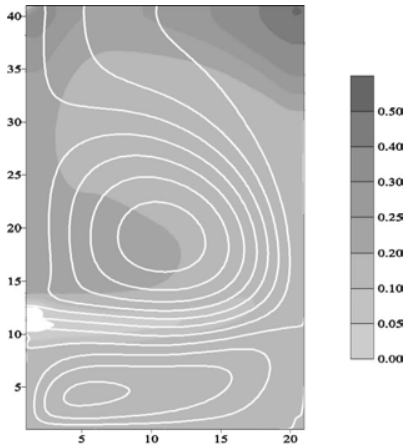


Figure 7 : Streamlines and temperature fields at L/B=2, H/h=14

As can be seen from Fig. 7 cold fuel oil enters into the tank from an inlet upper than the bottom of the tank. Two vortices form in the tank separated with a streamline just below the inlet level. An examination of the temperature results leads to the conclusion that the flow is relatively well-mixed in this case (*H/h*=14) compared to the first case (*H/h*=19). On the other hand, a stabilizing temperature gradient exists in a large portion of the tank in spite of the existence of low temperature values. This result is found to be compatible with the results shown in Fig. 3.

.

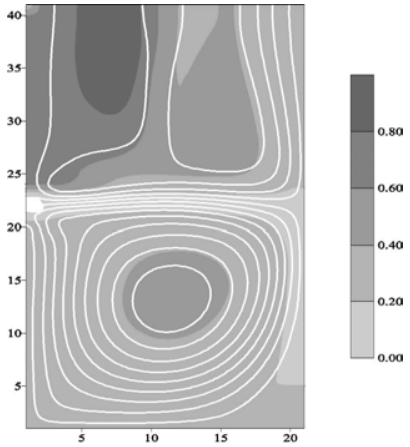


Figure 8 : Streamlines and temperature fields at L/B=2, H/h=9

The streamlines and temperature field in the case of H/h=9 are shown in Fig. 8. In this case a large vortex forms and the dimensionless temperature values are between 0.0 and 0.50 in the lower half portion of the tank. In the region above the inlet the flow does not seem to be a regular type and the dimensionless temperature variations are relatively higher than those in the lower half portion of the tank.

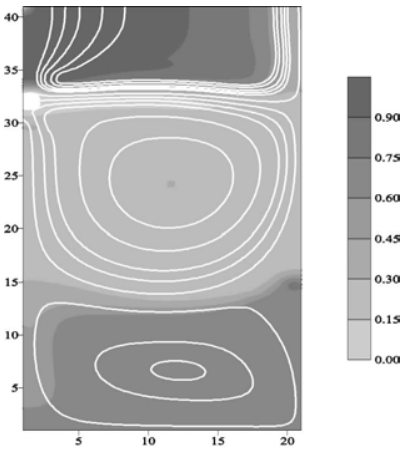


Figure 9 : Streamlines and temperature fields at L/B=2, H/h=9

As can be seen from Fig. 9 two large vortices form in almost whole of the tank in this case and the temperature field clearly seems to be irregular. The dimensionless temperature values at the top and at the bottom of the tank are higher than 0.6 while the dimensionless temperature values at the middle region of the tank are lower than 0.25.

Effect of aspect ratios on the transient average temperature variations

In order to investigate the effect of the aspect ratio on thermal behaviour of the tank the dimensionless transient average temperatures are plotted versus the dimensionless time, τ^*. For this aim the aspect ratio of the tank is now assumed to be *L/B*=1. The transient average temperature values for this type of tank are calculated for the same depths as the tank with the aspect ratio of *L/B*=2. The results for this case are given only for the cases of *H/h*=19 and *H/h*=14. As shown in Fig. 10 the transient average temperatures decrease slightly as soon as the lag time ends and then begin to increase slowly again for all depths. After τ^*= 0.78 the dimensionless average temperature values remain nearly constant at 0.87.

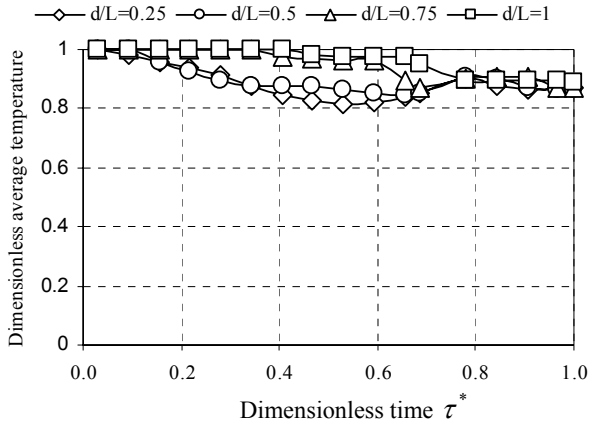


Figure 10 : Transient average temperature variations at four depths for *L/B*=1, *H/h*=19

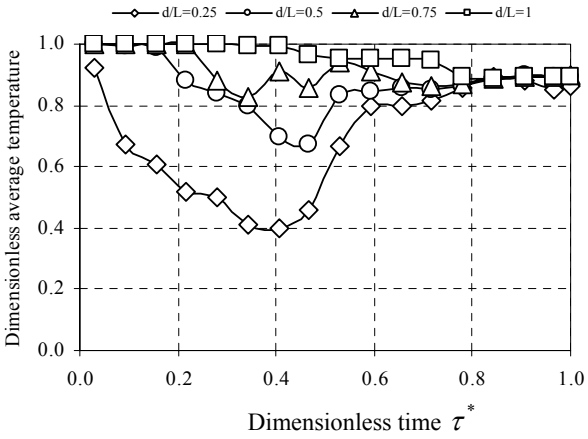


Figure 11 : Transient average temperature variations at four depths for *L/B*=1, *H/h*=14

In the case of *H/h*=14 the dimensionless average temperature variations at four depths of the tank are shown in Fig. 11. The lag time at *d/L*=0.25 is less than 0.005 and as soon as the lag time ends, the transient average temperature decreases rapidly from 1 to 0.4 at τ^*=0.4. The lag time is 0.15 for *d/L*=0.5 and the average temperatures for this depth decrease to 0.67 at τ^*=0.45. Then the average temperatures for these two depths begin to rise rapidly and after τ^*=0.8 they remain constant at 0.89. The average temperatures at the other two depths (*d/L*=0.75 and *d/L*=1) slightly decrease to 0.89 till τ^*=0.8 and then remain constant at this temperature level.

Conclusion

The effect of inlet position on transient turbulent flow in a fuel oil tank is investigated by a two-dimensional $k-\varepsilon$ model. The flow and temperature fields and the transient average temperature variations are presented for three different depths (d/L=0.25, d/L=0.5 and d/L=0.75) and fluid surface (d/L=1) in a tank with an aspect ratio of L/B=2. The average temperature values are found to be nearly homogenous at a dimensionless time of τ^*=1 when the inlet positions are H/h=14 and H/h=9 in the case of L/B=2. The streamlines and the temperature distributions are presented for four different inlet port locations in the case of L/B=2. Finally, the effect of aspect ratios on transient average temperatures in the tank are also investigated. The average temperature variations in three depths and fluid surface of a tank with an aspect ratio of L/B=1 are plotted versus the dimensionless times for two different inlet positions (H/h=19 and H/h=14).

References

[1] Turkish Loyd. *Storage of liquid fuels, lubricating, hydraulic and thermal oils, as well as oil residuals,* Rules for classification of steel ships, 1999. Chapter 4- Machinery , Section 10

[2] Alfa Laval Marine&Power AB. *Alcap Cleaning system for fuel oil*. 1999, Technical Information

[3] Altun M. *Chemical Tanker Operations*, 1999, (in Turkish)

[4] Gari HN. Loehrke RI. *A controlled buoyant jet for enhancing stratification in a liquid storage tank.* J. Fluid. Eng. 1982; 104: 475-481

[5] Baines WD, Martin WW, Smith DM. *Development of stratification in a rectangular tank by horizontal flow* J. Fluid. Eng. 1983; 105: 59-64

[6] Abu-Hamdan MG, Zurigat YG, Ghajar AJ. *An experimental study of a stratified thermal storage under variable inlet temperature for different inlet design* Int. J. Heat Mass Transfer 1992 ;35: 1927-1933

[7] Nakos, JT. *The prediction of velocity and temperature profiles in gravity current for use in chilled water storage tanks* J. Fluids Eng. 1994; 116: 83-90

[8] Homan KO, Soo SL. *Model of the transient stratified flow into a chilled-water storage tank.* Int. J. Heat Mass Transfer 1997; 40: 4367-4377

[9] Homan KO, Soo SL. *Laminar flow efficiency of stratified chilled-water storage tank.* Int. J. Heat Fluid Flow 1998; 19: 69-78

[10] Mo Y, Miyatake O. *Numerical analysis of transient turbulent flow field in a thermally stratified thermal storage water tank* Num. Heat Transfer 1996; 30: 649-667

[11] Cai, L, Stewart Jr. WE. *Turbulent buoyant flows into a two dimensional storage tank* Int. J. Heat Mass Transfer 1993; 36: 4247-4256

[12] Bates S, Morrison DS. *Modelling the behaviour of stratified liquid natural gas in storage tanks.* a study of the rollover phenomenon, Int. J. Heat Mass Transfer 1997; 40: 1875-1884

[13] Cotter MA, Michael EC. *Transient cooling of petroleum natural convection in cylindrical storage tanks-I. Development and testing of a numerical simulator.* Int. J. Heat Mass Transfer 1993; 36: 2165-2174

[14] Ziotko J, Supernak E, Mojzesowicz G. *Transport and storage of oil and some aspects of ecology* J. Construct. Steel Res. 1998; 46: 290

[15] Vardar, N. *Numerical analysis of the transient turbulent flow in a fuel oil storage tank* Int. J. Heat Mass Transfer, 2003; 46: 3429-3440

[16] Lage JE, Bejan A, Anderson R. *Removal of contaminant in a slot ventilated enclosure* Int.J. Heat Mass Transfer 1992; 35: 1169-1179

[17] Torii S, Yano T, Tuchino F. *Heat transport in a turbulent parallel Couette flows in concentric annuli for various Prandtl numbers* In: Proceedings of the 3rd Int. Thermal Energy Congress, Kitakyushu, Japan, 209-214

[18] Launder BE, Spalding DB. The numerical computation of turbulent flows, Comp. Meth. App. Mech. Eng. 1974; 3: 269-289

[19] Chow CY. 1979, An introduction to computational Fluid mechanics, John Wiley, Canada, 1979

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

Mechanical/Electronic Systems

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

Analysis of Noise in Ventilating and Air-Conditioning Systems

M. Nikian and M.Naghashzadegan*
Department of Mechanical Engineering
University of Guilan, Rasht, Iran

Abstract

A mathematical model was developed for the analysis of noise in ventilating and air-conditioning system. The model produces a static noise analysis, calculating sound level at an observer located in the room, considering the sound power created by a fan and the attenuation of this sound power through the system. Comparing the sound level with the maximum noise level allowed in the room, the amount of additional attenuation can be calculated.

Keywords: Noise Analysis; Computer software; Air Conditioning system.

Introduction

The air-conditioning system serving a room or space is frequently the major determinant of noise levels in a room [2, 3]. That is why a suitable acoustical environment is as important for human comfort as other environmental factors such as temperature and relative humidity. Finding an appropriate sound level for all activities rather than the lowest possible level is the goal of this research [1].

Doing this the system noise level first must be evaluated and then controlled to achieve a satisfactory acoustical environment in a room. Several paths are exist by which system noise reaches a listener. They are including airborne transmission of equipment noise to adjacent areas through the mechanical room construction; structure borne transmission of equipment vibration through the building structure; and duct borne noise created and transmitted by air-handling systems and their components.

*Corresponding author: Tel/Fax: (+98) 131-3232204 , E-Mail: naghash@guilan.ac.ir

Mathematical Model for Noise Analysis

The method of calculation for acoustic analysis in ventilating systems is described in various publications e.g. the ASRHAE guide [4], the CIBSE guide [5], or the “Design for Sound” [6] published by Woods of Colchester. The method described in each publication is essentially the same and is the method that will be applied throughout this project.

Objectives

The objectives of the project are therefore to produce a well structured model which can allow the user to input relevant details of fan, room, and connecting duct system to describe a particular section of air-conditioning or ventilating system.

Noise Calculations

The most reliable method of determining the amount of acoustic energy fed into the duct system is to get the fan manufacturer to provide sound power levels in each octave band of frequency. The levels should have been determined by a standard test method, preferably British Standard 848, Part2. If this data is available then it can be entered into the

program using the user input option from the Fan Menu.
An estimate of sound power generated can be made by using one of three empirical formulae, originally developed by Beranek [2]:

$$SWL = 67 + 10\log_{10}^{M} + 10\log_{10}^{h} \quad (1)$$

$$SWL = 40 + 10\log_{10}^{V} + 20\log_{10}^{h} \quad (2)$$

$$SWL = 95 + 20\log_{10}^{M} - 10\log_{10}^{V} \quad (3)$$

Where:
SWL = overall sound power level (dB)
V = delivered volume (l/s)
h = rated fan motor power (KW)

Any of three equations may be used, depending upon which details of the system are known.

Attenuation in the duct system

The following section will describe the attenuation calculation for the various duct elements.

Plain duct runs

The attenuation of lined ducts can be computed from a formula developed by Sabine [3]:

$$IL = \frac{P}{A}.l.a^{1.4} \quad (4)$$

where:

IL = insertion loss (dB)
P = perimeter of lined duct (m)
A = free cross sectional area of duct (m^2)
l = length of lined duct (m)
a = random incidence absorption coefficient of lining material when fixed to a rigid backing

Branches/Junctions

For most typical duct branches it can be assumed that the energy divides between the main and take-off ducts in the same way as the airflow does. That is, the total acoustic energy remains constant but is divided amongst the branches so that, in any one branch, the acoustic energy is less than that in the approach duct. The attenuation in any one branch is therefore given by:

$$10.\log_{10}\frac{\text{flow1}}{\text{flow2}} \quad (5)$$

where:
flow 1 = vol. flow of air in branch considered (l/h)
flow 2 = vol. flow of air up to branch (l/s)

Duct terminations

Noise of wavelengths that are long with respect to the dimension of the duct outlet tend to be reflected back within the duct rather than path into the room. Equations, for the relevant octave bands are given as:

$$63\text{ Hz}: IL = 3.73 - 7.46\log_{10}^{A} \quad (6)$$

$$125\text{Hz}: IL = 0.27 - 6.92\log_{10}^{A} \quad (7)$$

$$250\text{Hz}: IL = -1.61 - 5.20\log_{10}^{A} \quad (8)$$

$$500\text{Hz}: IL = -2.38\log_{10}^{A} \quad (9)$$

$$1000\text{Hz}: IL = 0.43 + 0.81\log_{10}^{A} \quad (10)$$

where:
IL = insertion loss (dB)
A = outlet area (m^2)

At higher frequencies i.e. above 1000 Hz, there is negligible end reflection and so it is assumed that there is zero attenuation.

Plenum chambers

In a room (as shown in Fig.1), the sound pressure level at the outlet point (and hence the sound power leaving) will compromise of the energy radiated to that point from the inlet, and the reverberant energy in the chamber.
The appropriate loss in sound power across the plenum is given by the equation:

$$SWL_1 - SWL_2 = 10.\log_{10} S_2\left(\frac{\cos p}{2PI\,d^2} + \frac{1}{R_c}\right) \quad (11)$$

where:

SWL_1 = sound power level entering chamber (dB)
SWL_2 = sound power level leaving chamber (dB)
d = slant distance from centre of inlet to centre of outlet (m)
p=angle *d* makes with the inlet axis (degrees)
R_c= "room constant" of plenum (m^2)

R_c is defined by the equation:

$$R_c = \frac{S_T . A_m}{1 - A} \qquad (12)$$

where:

S_T = total surface area of the plenum, including inlet and outlet areas (m^2)

A_m = average absorption coefficient of the internal surfaces of the plenum

A_m is defined as:

$$A_m = \frac{A . S_T + S_1 + S_2}{S_T} \qquad (13)$$

where :

S_1 = inlet area

S_2 = outlet area

S_L = Lined wall area

S_U = Unlined wall area

S_T = Total wall area = $S_1 + S_2 + S_L + S_U$

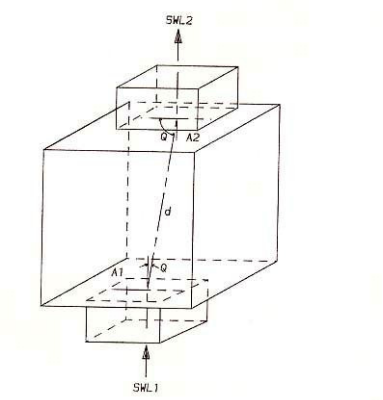


Figure 1: Plenum chamber Room sound Calculations

The relationship between the sound power level (SWL) of a source in an enclosed space whose acoustic properties are dictated by the room constant R_c , and the sound pressure level (SPL) at a point distance r and angle Q from the source is given by [2] :

$$SPL = SWL + 10 . \log\left(\frac{D_Q}{4 \pi r^2} + \frac{4}{R_c} \right) \qquad (14)$$

Direct room sound pressure level

The direct sound pressure level is obtained from the first term in the bracket of equation (14):

$$SPL_D = SWL - 20 \log_{10}^{r} + DI(Q) - 11 \qquad (15)$$

where:

SWL = sound power level at the duct outlet (dB)

r = dist. of the observation point in the room from the centre of the duct termination (m)

$DI(Q)$ = directivity index for angle Q between r and the duct axis.(= $10 \log_{10}^{D_Q}$) dB

D_Q = the directivity factor at angle Q

Reverberant room sound pressure level

The reverberant sound pressure level is obtained from the second term in the bracket of equation (14):

$$SPL_R = SWL - 10 \log_{10} R_c + 6 \qquad (16)$$

where:

SWL = the sound power level at the duct outlet (dB)

R_c = room constant, defined as:

$$R_C = \frac{S . A}{1 - A} \ (m^2) \qquad (17)$$

S = total surface area of the room (m^2)

A = average absorption coefficient of the room and is defined as;

$$A = \frac{S_1 A_1 + S_2 A_2 + S_3 A_3 + ... + \text{etc}}{S} \qquad (18)$$

Resultant room sound pressure level

The resultant room sound pressure level (SWL_T) is the logarithmically combined direct sound pressure level and direct sound pressure level, and is obtained from:

$$SWL_T = 10 \log_{10}\left(10^{SPL_D/10} + 10^{SPL_R/10}\right) \qquad (19)$$

where:

SPL_D = direct room sound pressure level (dB)

SPL_R = reverberant room sound pressure level (dB)

Maximum sound power level per terminal

The need for attenuation is normally assessed on whether the sound power level at a duct terminal will meet a specified noise rating in

the room. Permissible sound power at the terminals (SWL_t) is defined as:

$$SWL_t = NR + \text{room effect} \quad (20)$$

where:

NR = noise rating set for the room

Room effect = SWL leaving outlet – total room SPL

If there are a number of terminals in the room this gives the total sound power level from all the terminals. the maximum permissible sound power output for a single terminal (SWL_m) is equal to:

$$SWL_m = SWL_t - 10\log_{10} N_t \quad (21)$$

where:

N_t = number of terminals in room

SWL_t = permissible terminal sound power level

Required attenuation

The attenuation required in the system is defined as;

$$\text{Attenuation} = SWL_O - SWL_m \quad (22)$$

where:

SWL_O = sound power level at outlet (dB)

SWL_m =maximum sound power level per terminal (dB)

Verification Problem

In order to ensure that the Noise analysis program functions correctly and the subprograms interact correctly, an example calculation has been run through the program as well as being solved manually by the method described in previous part.

Figure 2 shows the ventilating system on which the sample calculations are to be based. The system is designed to distribute 5.1 l/s of air, at which volume flow the resistance of the system is expected to amount to 266 Pa. The laboratory is assumed to have a NR 45 noise rating and the distance to the nearest listener is 1.5m. The duct outlet area is 0.09m^2. the room volume is 308 m^2 and the reverberation time is taken as 1.75s.

Results

The results from the solution produced by the Noise Analysis model are shown in table 1 and 2. Figures 3 to 5 and Table 3 compare the calculations from the manual and mathematical analysis of Fan SWL leaving diffuser F, Reverberant SPL, Direct SPL, Resultant SPL, NR 45, Allowed SWL at terminal and Attenuation required in each octave band, that each solution has produced.

The results compare very well, thus verifying the mathematical model for noise analysis in ventilating system. By comparing the sound level values throughout the analysis with the corresponding values in the mathematical model, a very good match is found. These results verify that the “Noise Analysis model” data handling procedure and calculation procedure as well as the interaction between the various subprograms to be functioning correctly.

Conclusion

The research has involved developing a mathematical model to analyses noise in ventilating and air-conditioning systems. The first stage was to formulate a method of calculation by examining existing design procedures. The next stage was to set up a series of data bases. The mathematical model was based on a computer program. Finally the main stage was to write the actual Noise Analysis program. The main program consists of a series of separate program modules, each being responsible for the separate stages of noise analysis. Because of this modular structure the program can be easily modified and extended, which was one of the project objectives.

To verify that the program functions correctly, a verification problem was solved manually, using the calculation described. The problem was then solved using the “Noise Analysis Program”. The results of each solution compared very well. this indicates that the program functions correctly with respect to data entry/storage and calculation procedures.

References

1. Sharland I, “Woods Practical guide to Noise Control”, Woods Acoustics, 1979.

2. Woods, R. I., "Noise Control in Mechanical Services", Sound Attenuation Ltd. and Sound Laboratories Ltd., 1972.
3. Igbal, M. A., Willson, T. K. and Thomas, R. J., "The Control of Noise in Ventilation Systems- a designers guide", Atkins Research and Development, 1977.
4. "ASHRAE Guide and data Book Systems and Equipment", Chapter 52, American Society of Heating, Refrigeration and Air Conditioning Engineers, 1987.
5. "Design for sound', Woods of Colchester Ltd., 1970.
6. "CIBSE Guide", section B.12, Chartered Institute of Building services Eng., 1986.

Tables

	Octave Band (Hz)							
	63	125	250	500	1000	2000	4000	8000
Straight duct run (A-B)	80.6	78.6	81.8	82.4	81.4	78.4	73.4	71.4
Branch (B)	76.7	74.7	76.9	70.5	70.5	70.5	66.5	64.5
Straight duct run (B-C)	75.5	73.5	76.0	69.9	70.1	70.1	66.1	64.1
Junction (C)	72.5	70.5	73.0	65.9	59.1	60.1	59.1	58.1
Straight duct run (C-D)	70.5	68.5	71.5	64.9	58.1	59.1	58.1	57.1
Elbow (D)	70.5	68.5	71.5	63.9	50.1	52.1	54.1	54.1
Junction (E)	67.5	65.5	68.5	59.9	39.0	42.0	47.0	48.0
Straight duct run (E-F)	65.9	63.9	67.3	59.1	38.3	41.3	46.3	47.3
Duct termination (Diffuser F)	54.3	56.4	63.5	56.6	38.3	41.3	46.3	47.3

Table 1: Noise levels leaving each duct section (dB)

	Octave Band (Hz)							
	63	125	250	500	1000	2000	4000	8000
Input SWL from fan	83.0	81.0	83.0	83.0	82.0	79.0	74.0	72.0
Fan SWL at outlet	54.3	56.4	63.5	56.6	38.3	41.3	46.3	47.3
Reverberant room SPL	47.7	49.7	56.8	50.0	31.6	34.6	39.6	40.6
Direct room SPL	43.8	45.8	54.9	49.1	31.7	35.7	40.7	41.7
Resultant room SPL	49.2	51.2	59.0	52.5	34.7	38.2	43.2	44.2
NR 45 requirements	70.0	61.0	54.0	48.0	45.0	42.0	40.0	38.0
Allowed terminal SWL	76.0	66.0	58.0	52.0	49.0	45.0	43.0	41.0
Attenuation required	0.0	0.0	5.5	4.6	0.0	0.0	3.3	6.3

Table 2: Noise levels resulting in room (dB)

	Octave Band (Hz)							
	63.0	125	250	500	1000	2000	4000	8000
Computer analysis	0.0	0.0	5.5	4.6	0.0	0.0	3.3	6.3
Manual analysis	0	0	5.2	4.5	0	0	2.8	5.8

Table 3: Comparison the results of resultant attenuation required

Figures and Graphs

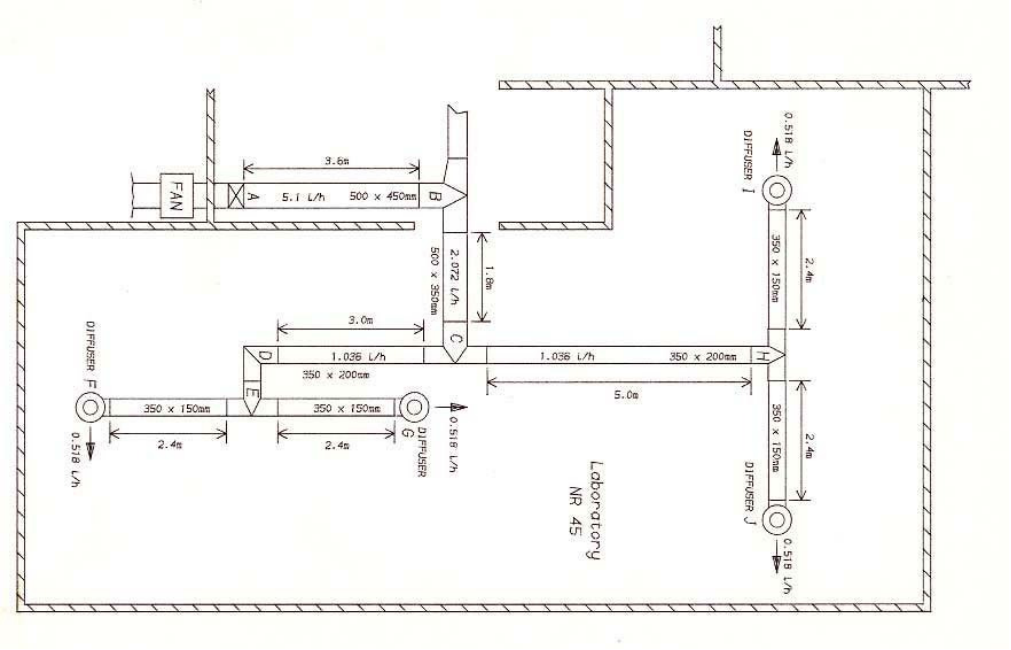


Figure 2: Example ventilating system

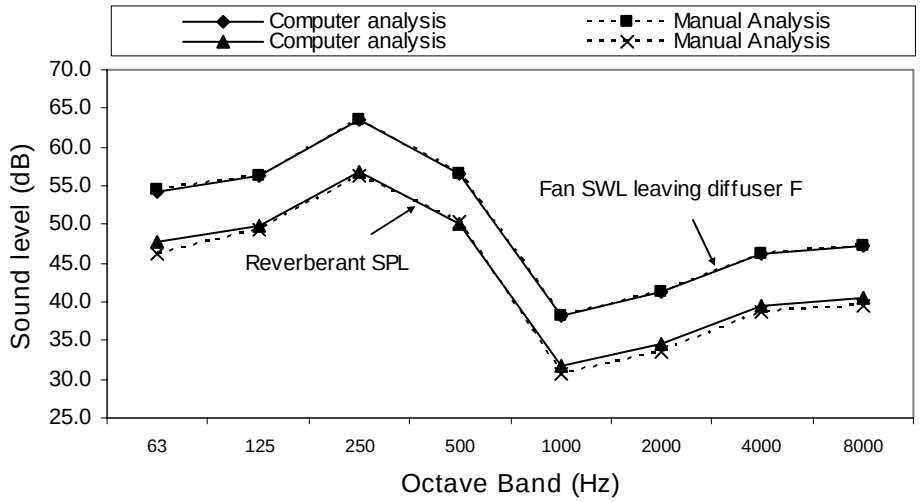


Figure 3: Comparison the results of Reverberant SPL and Fan SWL leaving diffuser

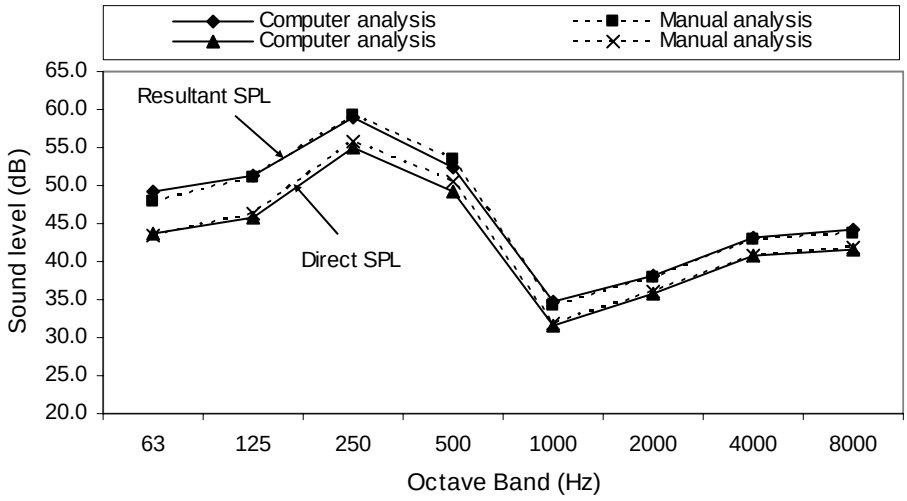


Figure 4:Comparison the results of Direct SPL and Resultant SPL

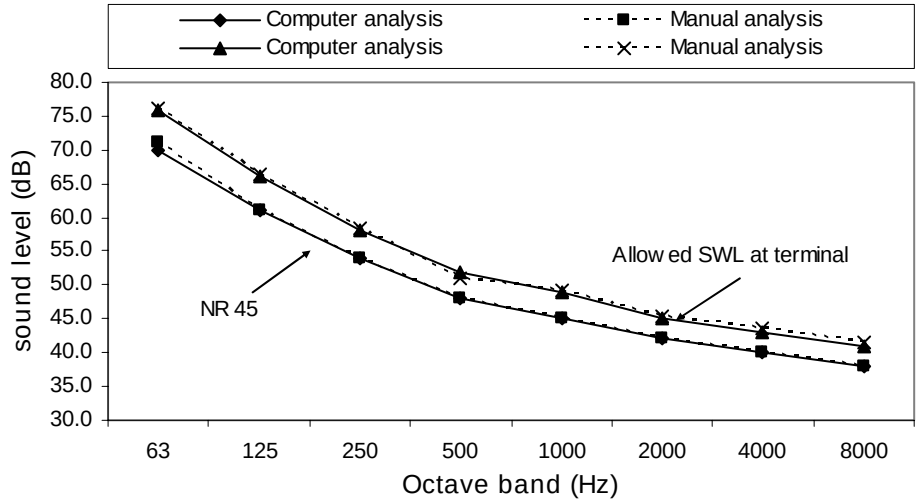


Figure 5: Comparison the results of NR 45 and Allowed SWL at terminal

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

Simulated Stress and Stretch of SWCNT

Ana Proykova* and Hristo Iliev
University of Sofia, Faculty of Physics
5 James Bourchier Blvd, Sofia 1126, Bulgaria

Abstract

The mechanical stability of open single wall carbon nanotubes (SWCNT) under axial stress (compression and tension) and twist has been re-examined in a search of specific tube-length and load scaling. SWCNT with different chiralities and lengths have been simulated with a classical molecular dynamics method employing the many-body empirical Tersoff-Brenner potential. Stress has been achieved by enforcing constant linear velocity on the edge atoms from both sides of the SWCNT as suggested by Srivastava and Barnard. We have found opposite length scaling at fast (1/10 of v_s the sound velocity in carbon tubes) and slow (1/20 v_s) loading of (10, 0) tubes. Another finding is that at fast loading short zigzag (10, 0) tubes transform from elastic to plastic states before they break in the middle, while tubes, longer than 13 nm, break-up directly in the elastic state. Thus, short tubes behave like metals or ionic solids, while long tubes resemble ceramics or glasses under the conditions studied. All tubes form spiral-like structures when twisted. Standing waves, generated under specific conditions, determine the different behavior of tubes with various lengths and chiralities.

Keywords: Molecular Dynamics simulations, elasticity and inelasticity, single-wall carbon nanotubes, nanoscale pattern formation, constant-composition solid-solid phase transformations

PACS: 02.70.Ns, 81.30.Hd, 81.40.Jj, 81.40.Vw

Introduction

Since their discovery [1], single-wall carbon nanotubes (SWCNT) have been considered as members of the fullerene family [2], having a high aspect ratio, a few defects, and unique mechanical and electronic properties. Their mechanical properties, reviewed in [3], are closely related to both electron conductivity [4] and adsorption [5]. Hence, mechanical loads on SWCNTs can be used to design a good dynamical chiller, based on adsorption–de-sorption of water molecules. This possibility of engineering justifies new simulations of loading, including twist of tubes, which is motivated by a possible application of nanotubes as shafts in nano-electromechanical devices.

Nanotubes are quasi-3D cylindrical objects made of rolled-up graphite sheets, Fig.1. The vector $\mathbf{C}_h = n\mathbf{a}_1 + m\mathbf{a}_2$ (n, m); n, m - integers, connects crystallographically equivalent sites on the sheet. The angle between $\mathbf{C}_h$ and the zigzag line (n,0) specify the tube type: (2n,n) is chiral, (n,n) is armchair, (n,0) is zigzag. The tube diameter is computed as follows: $d=0.078\ (n^2 + m^2 + nm)^{1/2}$ *nm.* The tube curvature and chirality determine the SWCNT conductivity. The zigzag (n,0) SWNTs should have two distinct types of behaviour: the tubes will be metals when $n/3$ is an integer, and otherwise semiconductors. However, the tube (5,0) with $d < 4$ is metallic, which should be related to the band-gap change due to the curvature [6].

* Corresponding author. Phone: +359 2 8161828, Fax:+359 2 962 52 76, E-mail: anap@phys.uni-sofia.bg

This article presents evidences for different responses of tubes with different lengths to applied stress and twist: short (l < 13 *nm*) zigzag (10,0) tubes sustain a large amount of compression and recover their initial shape when the external force is set to zero. The longer tubes fold irreversibly under the same force (strain energy per atom).
We simulate the behavior of open-ended SWCNTs (no periodic boundary conditions) under heavy loading. We consider zigzag (10,0), armchair (6,6), and (10,10) tubes with lengths between 8,63 *nm* and 54,3 *nm*. The aim is to answer the following questions: What is the amount of stress (twist) necessary to *break bonds*, i.e. elastic-to-plastic transformation of a SWCNT with a given length and chirality? What is the change of the potential energy surface (PES) under such extreme loading?

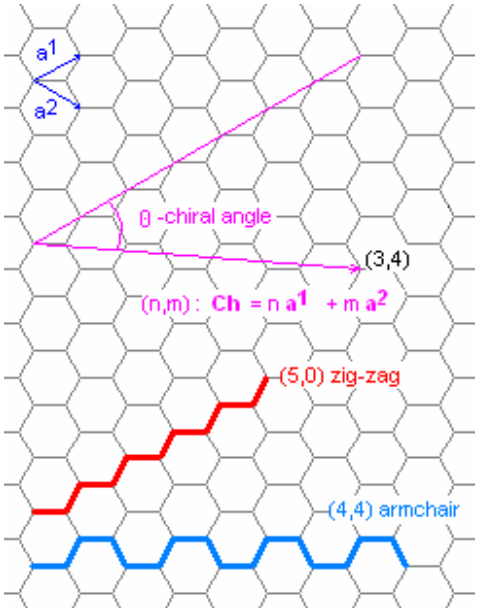


Fig.1 Rolled-up graphite sheets make tubes.

The first question arises in relation of elastic-to-plastic transition to the internal atomic response to an externally applied stress. The carbon atoms in tubes are covalently bonded - the stiffest bond that determines the exclusive mechanical stiffness of carbon nanotubes. However, a given material has two kinds of stiffness: one for a fast loading, when there is too little time for relaxation, 'unrelaxed' process, and another for slow loading which allows relaxation to occur, relaxed process. In solids with dense packing these two do not differ much. Here we show how carbon nanotubes behave under fast and slow loading.
The second question is related to physisorption: a PES with many local minima is a better adsorbent than an even one. Broken bonds, due to high stress, facilitate creation of adsorbing areas in PES.
Up to our knowledge, the scaling of tension induced plasticity of carbon nanotubes is studied here for the first time.

Model for mechanical loading

To model formation of carbon tubes, we solve the Newtonian equation of motion $m_i \ddot{\mathbf{r}}_i(t) = -\nabla_{\mathbf{r}_i} E_b$, with E_b being a simplified form of the Brenner-Tersoff potential [7]:

$$E_b = \sum_i \sum_{j(>i)} \left[V_R(r_{ij}) - \bar{B}_{ij} V_A(r_{ij}) \right], \quad (1)$$

i and j run over all atomic sites; r_{ij} is the distance between i-th and j-atom; the repulsive term V_R is

$$V_R(r_{ij}) = f_{ij}(r_{ij}) D_{ij}^{(e)} / (S_{ij} - 1) e^{-\sqrt{2S_{ij}}\beta_{ij}\left(r_{ij} - R_{ij}^{(e)}\right)} \quad (2)$$

and the attractive terms V_A is

$$V_A(r_{ij}) = f_{ij}(r_{ij}) D_{ij}^{(e)} S_{ij} / (S_{ij} - 1) e^{-\sqrt{2/S_{ij}}\beta_{ij}\left(r_{ij} - R_{ij}^{(e)}\right)}, \quad (3)$$

with $D_{ij}^{(e)}$, S_{ij}, β_{ij} and $R_{ij}^{(e)}$ - parameters given in Table 1.The function f_{ij}, which restricts the pair potentials V_R and V_A to nearest neighbors is:

$$f_{ij}(r) = \begin{cases} 1, r < R_{ij}^{(1)} \\ \left[1 + \cos\left[\dfrac{\pi\left(r - R_{ij}^{(1)}\right)}{R_{ij}^{(2)} - R_{ij}^{(1)}} \right] \right] / 2, R_{ij}^{(1)} < r < R_{ij}^{(2)} \\ 0, R_{ij}^{(2)} < r \end{cases}$$

$\bar{B}_{ij} = \left(B_{ij} + B_{ji}\right)/2$ is an empirical bond-order function; each of the averaged terms has the form:

$$B_{ij} = \left[1 + \sum_{k(\neq i,j)} G_i(\phi_{ijk}) f_{ik}(r_{ik}) e^{\alpha_{ijk}\left[(r_{ij} - R_{ij}^{(e)}) - (r_{ik} - R_{ik}^{(e)})\right]} \right]^{-\delta_i}$$

For carbon atoms (index 'C'), the function G_C is:

$$G_C(\phi) = a_0 \left\{ 1 + c_0^2 / d_0^2 - c_0^2 / \left[d_0^2 + (1 + \cos\phi)^2 \right] \right\}$$

ϕ is the angle between the lines, connecting i-th with j-th atom and i-th with k-th atom. The potential parameters have been adjusted [7] to reproduce the bonding structure of graphite, diamond, carbon nanotubes and small hydrocarbon molecules.

Parameter	Value	Parameter	Value
$R_{ij}^{(e)}$	1.39	$R_{ij}^{(1)}$	1.7
$D_{ij}^{(e)}$	6.0 eV	$R_{ij}^{(2)}$	2.0
β_{ij}	2.1 $^{-1}$	a_0	0,00020813
S_{ij}	1.22	c_0^2	330^2
δ_{ij}	0.5	d_0^2	3.5^2
α	0.0		

Table 1: Potential parameters used in simulations.

We use the Molecular Dynamics (MD) method [8], which is a numerical technique for integration of ordinary differential equations. In the presence of an external field ***F*** (stress or twist) applied to a collection of atoms, the Newton's second law is:

$$m_i \ddot{\mathbf{r}}_i(t) = -\nabla_{\mathbf{r}_i} E_b + \mathbf{F}_{ext}(t), \qquad (1)$$

m_i is the i-th atom mass. Integration is performed with the velocity Verlet algorithm [8], which advances in time the particle positions and velocities as follows:

$$\begin{vmatrix} \mathbf{r}_i(t+dt) = \mathbf{r}_i(t) + dt.\mathbf{v}_i(t) + (2m)^{-1} dt^2 \mathbf{F}_i(t) \\ \mathbf{v}_i(t+dt) = \mathbf{v}_i(t) + (2m)^{-1}[\mathbf{F}_i(t) + \mathbf{F}_i(t+dt)] \end{vmatrix},$$

where dt is the time step (0.2 fs in our simulations). The dt value satisfies two requirements: a) energy conservation $\Rightarrow$ dt must be small; b) reversibility of the classical trajectory if dt - dt, $\Rightarrow$ dt must be large enough to reduce the round-off errors accumulated in very long runs needed for a very small step. The value of dt =0.2 fs ensures the energy conservation (in a constant energy run, no external field) up to 10^{-5} eV/atom. The cell linked-list method and the Verlet neighboring list [8] have been combined to speed-up the calculations [9].

The tube starting configurations – zigzag, chiral, or armchair - are generated with the Mintmire's code [10] and consequently relaxed (optimized) by a power quench in a MD run [11]. In the power quench, each velocity component is set to zero if it is opposite to the corresponding force of that component. This affects atomic velocities, or unit-cell velocities (for cell shape optimizations).

The stress is simulated by changing at each time step only the positions {**r**} of the edge atoms of the open-end tube, {**r**} {**r** ± **r**} [12]. Hence, the edge atoms do not move according to the classical equation of motion, while positions and momenta of the rest of the atoms are computed from Eq. 1. The coordinate system is centered in the middle of the tube with the z axis along the tube length. Thus, for compression the sign is (+) for the left side atoms and (-) – for the right side atoms. For tension, the signs are altered. Various values of constant velocity of the edge atoms have been tested to check the nanotube responses to fast and slow loading. Here we report results for 500 and 1000 m/s, e.g. 1/20 and 1/10 of the sound of velocity in carbon. The stress waves, corresponding to these velocities, cause (or not) a generation of standing waves depending on the tube length. Due to the standing waves and beating, some atoms of the tube do not displace while vibrate with maximum amplitudes, which produce bond-breaks.

To be exact, the stress at a point is determined by $\sigma = \lim_{\Delta A \to 0} \frac{\Delta P}{\Delta A}$, where P is the load being carried by a particular cross section A. In general, the stress may vary from point to point as it is in the case of non-equilibrated tubes we study. The stress has two components, one in the plane of the area A, the *shear stress*, and one perpendicular, the *normal stress*. It is always possible to transform the co-ordinates on the body into a set in which the shear stress vanishes. We consider the remaining normal stresses called the principal stresses. The stress cannot be measured directly but is usually inferred from measurements of strain, i.e. a change

in size and/or shape. *Stretch* is the change in length: $e = \frac{\Delta l}{l_0}$, where l is the change in length and l_0 is the original undeformed length. In tension $e>0$, i.e. the body has been lengthened; $e<0$ in compression.

The twist is realized as follows: at every time step we rotate in the opposite directions the edge atoms around the tube axes at angles between 0.001 and 0.00001 rad/s. Depending on the velocity of rotation standing waves again appear.

To study the effect of the standing and reflected waves we applied the external force at times 0, , 2 ,…:

$$m_i \ddot{\mathbf{r}}_i(t) = -\nabla_{\mathbf{r}_i} E_b + \sum_{n=0} \delta_{t,n\tau} \mathbf{F}_{ext}(t), \qquad (2)$$

where $\delta_{t,n\tau}$ is the Kronecker delta. For large values of (>2 ps), the tubes have time to relax between the successive loads and obey the Hook's law.

Results

Defect formation (bond breaking, vacancy), energy accumulation, and shear of the tubes have been monitored during the calculations.

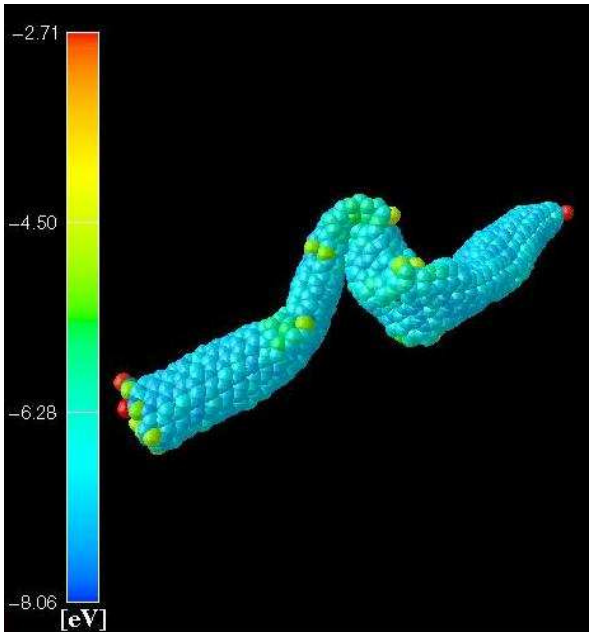


Figure 2: A short tube, (10,0) 12.98 *nm*, bends under compression.

Our results for uniaxial stress are consistent with the data published [12] for (10, 10) tubes with different lengths. Short tubes (up to 13 *nm*) sustain large amount of compression (~ 30%) without breaking and almost completely recovered their initial shape when the external force is set to zero. Under compression several atoms reside off-plane Fig.2 and can only break one or two bonds. Less bounded atoms (lower binding energy) are green in the Figure 2. The red atoms are 'hot' and occasionally sublimate. If the compression time is short, the pressure is below 100 GPa, the tube completely restores its initial shape when released; otherwise it buckles. Salvetat and co-workers [3] have already shown in their experiments that the tubes are elastic for low loads. The elastic regime is proper for periodic adsorption/de-sorption of noble gases, which do not form a chemical bond with the dangling bonds. However, the rate of adsorption is too low because the number of defects created in compression is small – green-yellow spots in the figure.

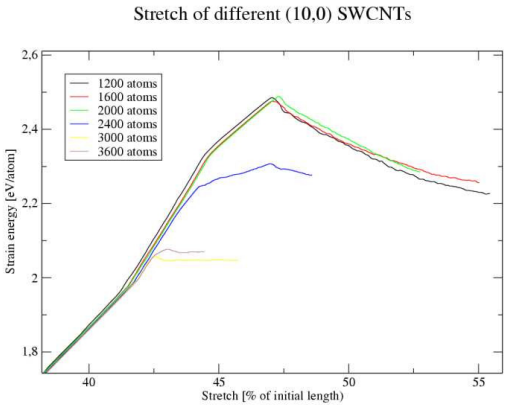


Figure 3: In tension two kinks of the curves are seen for short tubes; one kink - for long tubes. The maximum of the curves in all cases correspond to the break of tubes. The strain energy per atom rapidly reduces after the break.

Tension of the tubes, Figure 3, generates more defects then compression. Under tension, the tubes break for 42-44% extension of the different tubes studied here. This high percentage of extension is due to the covalent bonding of the carbon atoms and hexagonal structure of the graphene sheets. Inspecting the curve slopes, we notice two kinks

for (10,0) tubes with lengths less than 26.03 *nm* (2400 atoms). We identify the first kink as transition between the elastic and plastic states with broken bonds that do not re-connect when the external force is set to zero. These dangling bonds fluctuate slowly and attract noble gases.
We relate the second kink with the appearance of a 'bridging' area in the middle of tubes shorter than 26,03 *nm* like in the Figure 4. The tube is in a plastic state. We call a state 'plastic' if the system does not restore its initial shape after the force is set to zero.
Bonds break when enough energy is accumulated under tension. The time t (in *ps*), needed for a tube to gain energy for elastic-to-plastic transition, depends linearly on the tube length. The slow tension is given with $t(N) = 0{,}0102\ N - 1{,}6626$ and the fast tension is $t(N) = 0.0045\ N + 0.2612$; N is the number of carbon atoms. The different slopes point to different redistribution of the energy in slow and fast processes.

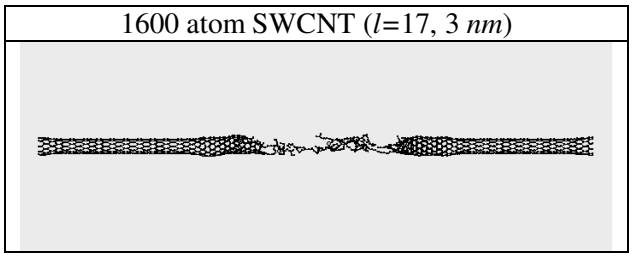


Figure 4: Broken bonds in the middle of the tube – a plastic state, corresponding to the second kink of the curve in the Figure 2.

Long tubes form 'bridges' nearby the two ends, Figure 5, and break directly from the elastic state.

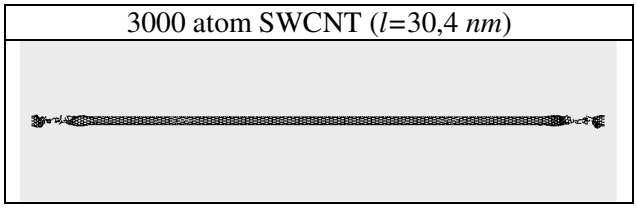


Figure 5: Long tubes break in the elastic state: no elastic-to-plastic transition occur.

These results indicate a size effect: under tension, *short tubes behave like metals*, while long tubes resemble ceramics under mechanical loads. The reason is the energy distribution along the tube axis and the stress wave propagation, which we study in detail for fast and slow processes.
We report here different responses of the tubes to slow and fast tension: the longer the tube the larger extension before break-up under low speeds, while at the faster velocity, the length scaling is opposite.

zigzag	No. of atoms	Tension velocity	Tension time	% stretch
(10,0)	1200	500m/s	10,36ps	43,8
(10,0)	1200	1000m/s	5,60 ps	43,1
(10,0)	1600	500 m/s	15,28ps	44,1
(10,0)	1600	1000m/s	7,52 ps	43,4
(10,0)	2000	500 m/s	18,20ps	44,2
(10,0)	2000	1000m/s	9,40 ps	43,3
(10,0)	3000	500 m/s	29,00ps	44,5
(10,0)	3000	1000m/s	13,76ps	42,3

Table 2: Fast and slow tension scale oppositely.

Fast loading, i.e. unrelaxed stress process, causes elastic-to-plastic transition in short tubes before they break. Slow loading does not initiate elastic-to-plastic transition for any tube length studied here. We explain this with the tube- and the stress-wave lengths becoming incommensurate.

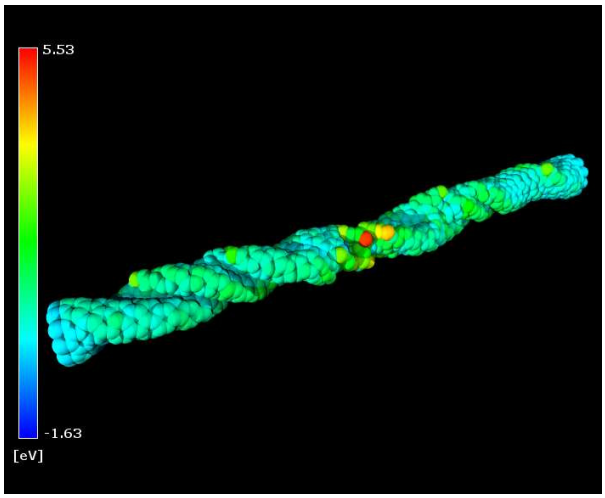


Figure 6: The red and yellow atoms are less bound in a twisted (10,0) tube with l=12,9 *nm*.

The scale in the Figure 6 corresponds to the difference between the initial energy and the current energy of every atom. The snapshot corresponds to 3,4 *ps* duration of twist. When off-line atoms occur, the twisted tube can not restore its original shape even if the external force is set to zero. We have periodically applied and released the twisting external force on the tube to study the process of energy accumulation and standing wave formation as a function of the period , Figure 7.

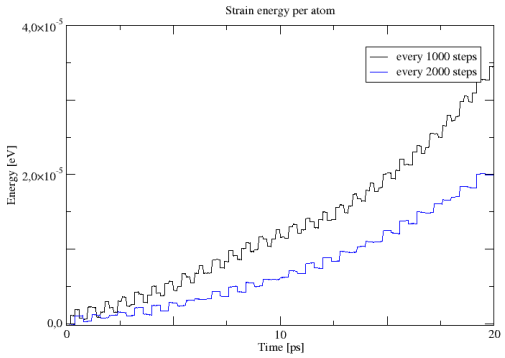


Figure 7: Accumulation of strain energy in a tube as a function of the period in Eq.2. Beating is well seen for =2000 steps = 0.4 *ps*: at 6, 10, 14, 18 ps.

The standing wave (beating) is a result of modulation of the stress-wave by reflected waves. In order to monitor the motion of each atom due to the standing wave, we perform Fourier transform of the atom kinetic energy. The Figure 8 shows a typical pattern of beating (~1.8 10^{13} Hz) observed for a single atom located near the tube end. The pictures for atoms, located at different sites of the tube, are similar, although the peaks positions could be shifted due to the resonance conditions.

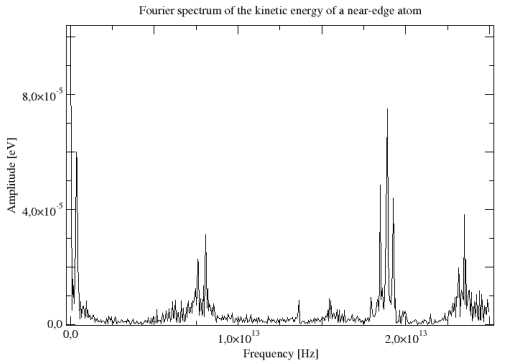


Figure 8: Beating is clearly seen in the Fourier spectrum of the kinetic energy of a single atom, 3th ring from the edge atoms.

IV. Conclusions and comments

This study shows that a relatively small number of dynamical defects can be created in an initially perfect SWCNT before it breaks-up. This is discouraging for the producers of defected nanotubes considered more interesting for nano-electronic applications. The good news is that once created, these defects do not migrate: a detailed study of defect formation and diffusion will be published in [15]. The atoms, surrounding a defect, vibrate and might become dynamical adsorption centers, which will be studied with the code based on the density-functional theory [13]. Our current calculations are limited as they are based on classical approximations.

In practice, the SWCNTs grow in ropes or bundles hence such computations could be of interest as well. However, these computations are expensive and require intelligent and new approaches. Possible solutions are the usage of clusters of computers and/or parallel computations as we have already demonstrated [9]. Our algorithm scales with the number of atoms N in a system as O(N).

We show that fast tension causes metal-like mechanical behavior of short tubes – they firstly transform from elastic to plastic state before break. Long tubes resemble ceramics or glasses under fast stress, i.e. break directly in the elastic state. This result should be remembered when designing nano tube devices.

Under tension, the atoms increase their temperature (zero in the beginning of the simulations). One could think of temperature induced transformations and related the specific size effects in mechanical load with the temperature-driven solid-solid changes in clusters of rigid molecules. These clusters exhibit various structural phase changes depending on the topography of the potential energy surface (PES) and particularly on the distribution of the saddle

points on the PES [14]. Special research of the potential energy topography of elongated and twisted tubes will be performed in the future to study unrelaxed processes.

Acknowledgements
This research was supported by the Ministry of Education and Science (Bulgaria) under the grant (F-3/2003). H. Iliev acknowledges the support from the Edinburgh Parallel Computing Center during his visit in 2003 {EC contract No.HPRI-CT-1999-00026 TRACS-EPCC}. A.P. is very grateful to Prof. Jeffery Gordon (Ben-Gurion University of the Negev, Israel) and Dr. Chua (the University of Singapore) for their valuable ideas, inspiring discussions, and stimulating experimental work in the field of adsorption of carbon tubes.

References:

[1] Iijima S. *Helical microtubules of graphitic carbon*. Nature 1991;354:56-58.

[2] Dresselhaus MS, Dresselhaus G, Avouris P. *Carbon Nanotubes: Synthesis, Structure, Properties, and Applications*. Berlin: Springer, 2001.

[3] Salvetat-Delmotte J-P, Rubio A. *Mechanical properties of carbon nanotubes: a fibre digest for beginners*. Carbon 2002;40:1729-1734.

[4] Adams II TA. *Physical Properties of Carbon Nanotubes: Equilibrium Structure*. 2003. http://www.pa.msu.edu/cmp/csc/ntproperties/equilibriumstructure.html

[5] Zhao J, Buldum A, Han J, Lu J.P. *Gas molecule adsorption in carbon nanotubes and nanotube bundles*. Nanotechnology 2002;13:195-200.

[6] Yang L, Anantram M, J. Han J, Lu J.P. *Band-gap Change of Carbon Nanotubes: Effect of Small Uniaxialand Torsional Strain*, Physical Review B 1999; 60(19): 13874-13878.

[7] Brenner DW. *Empirical potential for hydrocarbons for use in simulating the chemical vapor deposition of diamond films*. Phys. Rev. B 1990;42:9458.

[8] Allen MP, Tildesley DJ. 1986. Computer simulation of liquids. Oxford Science Publications.

[9] Iliev H, Proykova A. Applied Nanoscience 2004 (subm.)

[10] Mintmire J. *make_tube.f*. 2000; www.mse.ncsu.edu/CompMatSci/make_tube.f

[11] Ordejón P, Sánchez-Portal D, Artacho E, Soler JM, García A.1999. *SIESTA 0.12.7 USER'S GUIDE*. http://dipc.ehu.es/arubio/siesta/siesta.html

[12] Srivastava D, Barnard ST. *Molecular Dynamics Simulation of Large-Scale Carbon Nanotubes on a Shared-Memory Architecture*. Super Computing 97, 1997. Accessed 02.08 2004: http://www.nas.nasa.gov/Groups/Nanotechnology/publications/1997/sc97/INDEX.HTM

[13] Proykova A, Pisov S, Dragieva I, Chelikowsky J. 2002. DFT Calculations of Optimized Structures of Some Coordination Compounds. in *Nanoscience & Nanotechnology.*; 3:17-20. eds. E. Balabanova and I. Dragieva. Sofia: Heron Press Science Series.

[14] Proykova A. 2002. The Role of Metastable States in the Temperature Driven Phase Changes of Molecular Nanoclusters. *Vacuum* 69:97-104.

[15] Iliev H, Proykova A. Meeting in Physics ed. A. Proykova (accepted: v.5, 2004)

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

Magnetorheological Dampers in Earthquake Protection : a Simulation Study

Charles Stammers[1] and Emanuele Guglielmino [2]

Department of Mechanical Engineering, University of Bath
Bath BA2 7AY, UK

Dina D'Ayala
Department of Architecture & Civil Engineering,University of Bath

Tudor Sireteanu and Danut Stancioiu
Institute of Solid Mechanics, Romanian Academy, Bucharest, Romania

Abstract

An investigation is made of the feasibility of employing magnetorheological dampers for seismic protection of buildings. These require only a low power electrical actuation system and are inherently fail-safe as in the event of a power failure they become conventional passive viscous dampers. A robust control logic based on variable structure control theory is advanced. The response of a multi-storey building modelled as three blocks is analysed via computer simulation. The MR damper characteristics have been experimentally identified. Three systems of graduated complexity are examined: base sliding, sliding with adaptive damping, and adaptive damping with spring cancellation. A base clearance is included to represent a good passive protection system. With a peak input acceleration of 0.68g, drift at one-third height is reduced by a factor of three compared to the passive case. *Keywords:* magnetorhoeological damper, variable structure control, earthquake, simulation.

1. Notation

B_j	damping constant, block j	K_1	base stiffness
c_{jk}	damping constant	K_b	buffer stiffness
cl	base clearance	K_j	structural stiffness j =2, 3
c_0	MR damping coefficient	K_0	MR damper elastic coefficient
f_c	MR damper friction force	M_j	mass of block j
f_0	MR damper offset force	u	MR damper control current
F_b	buffer force	x_j	lateral displacement of block j
F_j	control force, damper j	z	ground lateral displacement
F_s	control force to lock sliding element	ζ_{soil}	soil damping ratio
h	storey height	ζ_{str}	structural damping ratio
H	height of building	ω_j	natural frequency (rad/s) sub-system j.

[1] Corresponding author. Phone: +44 1225 385962, fax +44 1225 826928

[2] Now at GE Energy, Florence

2. Introduction

The response of a structure to a seismic input can be controlled by placing damping elements in parallel with the load bearing structural system, assumed to behave elastically, between storeys and/or between the building and its foundation. This work investigates the feasibility of using magnetorheological (MR) dampers for this purpose.

A variety of techniques and control algorithms have been investigated over the years in order to control structural seismic response. Kobori *et al.* [1993] proposed a method based on variable stiffness control**,** while Yamamoto and Aizawa [1994] worked on an active mass damper. Controlled friction forces can be used to limit building response to an earthquake. A friction pendulum system has been proposed by Tsopelas *et al.* [1996]**.** Feng[1993, 1994] used sliding bearings to achieve base isolation. Such a system requires there to be a modest gap between the building and the ground. (see Figure 1) to allow some lateral motion of the building relative to the foundation. In Feng's study the friction force was adjusted by varying the pressure of the fluid in the bearing. In this way a low force can be used to isolate for small earthquakes and a larger value used to limit relative displacements during more severe earthquakes, when to avoid impacts with the foundation the relative motion between base and ground must be limited to the width of the gap or clearance (typically about 0.3 m - 0.4 m). In simulations based on a four-storey building with a sliding base, the predicted acceleration of the top storey in the active case was 40% of that in the passive case. Nishitani *et al.* [2000] carried out scale model experiments utilising variable friction dampers. A plate was held by an air-regulated clamp. Controllable friction braces have been studied by Akbay and Aktan [1991] and also by Dowdell and Cherry [1994]. Likewise a variety of algorithms has been proposed over the years. These include sliding mode control [Singh *et al*., 1996] reliability-based control [Field *et al.*, 1996], fuzzy control [Casciati *et al.*, 1997] and other non linear control techniques. A survey can be found in Spencer *et al.* [1998]. Stammers *et al.* [1999] carried out experimental work using a servo driven semi-active friction device to control a sprung mass excited by a shake table. The algorithm employed was designed to cancel spring forces whenever possible. This approach the authors call "balance" logic. The acceleration in the semi-active control case was 60% of that in the passive case, which compares well with the active control work of Feng. Further work in this area was reported by Guglielmino *et al.* [2000]. A spring cancellation control algorithm using semi-active control of a friction damper has been applied by the authors to buildings subject to seismic excitation [Stammers and Sireteanu, 1998, 2000a, 2000b]. The latter paper concerned the use of three semi-active devices (positioned at the base, at one-third height, and two- third height). The predictions were that both inter-storey drift and lateral accelerations can be reduced to about half the values in the passive case. The most effective damper was the one at the base.

Recently attention has been directed to the use of MR dampers which are easily controlled, provide high force at low power and offer good reliability. A MR damper is a device consisting of a fixed-orifice damper filled with a MR fluid having magnetic coils mounted on the piston head. A MR fluid is a particular type of oil which contains in suspension micron-sized particles of ferromagnetic material. By applying a magnetic field (via a current supplied to the coils) to the fluid, ferromagnetic particles form chains, thus changing the value of yield stress. As a consequence the rheological properties of the MR fluid change and the fluid passes from liquid state to semi-solid state. Hence by controlling the current to the coils it is possible to produce variations in the damping force. This creates a controllable damping action, without taking recourse to valves or any moving mechanical part. An overview of MR fluids can be found in Agrawal *et al.* [2001].

MR dampers are fail-safe. Damping is generated even in the passive mode. The mechanical system is highly reliable. Reliability has been proven already in the car industry where MR damper-based suspensions have been already in the marketplace for years on high class, high performance vehicles (e.g. Cadillac Imaj, Chevrolet Corvette). The MR damper requires only a battery and not mains power (which is often lost in an earthquake).

With recharge/replacement of the battery at regular intervals reliability should be high. With regard to durability of the MR damper itself, there are at least two situations which lead to limited life. One is over-use, as may occur in a motor vehicle application. This has been studied by two of the authors. The other situation is prompted by under-use. If the MR fluid is not agitated the ferromagnetic particle separate out within a matter of hours. Shaking recreates a suspended mixture within a second or so. However, if the damper were unused for a period of months or years, as is likely in an earthquake system, it is quite possible that the fluid would not reconstitute. It is therefore proposed that a pair of dampers are used alternately, with the spare one being shaken

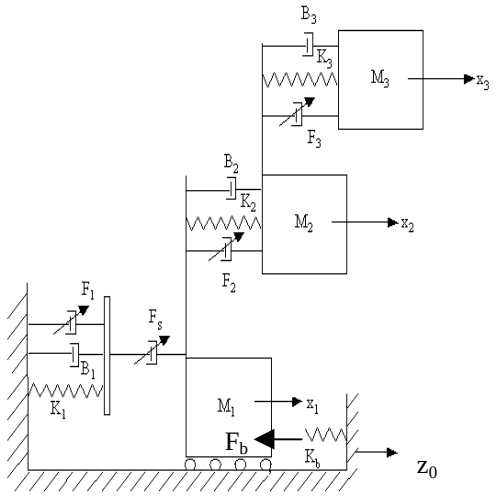
occasionally by means of an offset mass powered by an electric motor. The dampers would be changed over at regular intervals.(daily seems advisable) Dyke *et al.* [1996] proposed the use of an MR device employing a clipped optimal strategy based on acceleration feed back. A commercial damper with a gas reservoir was used.

The rise time was found to be only 8 ms. Simulations indicated that using the damper in the 'passive-on' condition (constant applied voltage yielding maximum damper force) roughly halved inter-storey drift and lateral accelerations.

Passive-off (with the damper operating purely as a viscous device) produced similar results. Clipped optimal control strategy reduced drift further, but at the expense of greater lateral accelerations than achieved by the passive-on system.

3. Model of the Building and of the MR Damper

In this study the building is represented as three blocks (see Figure 1). The blocks do not have to be the same mass, so any number of floors can be represented. The MR dampers are represented by variable damper symbols. Figure 1 indicates a model which includes a further sliding element in series with the base control unit. In the work reported here the sliding element is locked and F_s does not appear in the equation of motion. Sliding occurs between the base mass M_1 and the ground.



Semi-active control scheme

The seismic input is denoted by z_0

.

Figure 1 Building model

The equations of motion are:

$$M_1\ddot{x}_1 = -K_1(x_1 - z_0) - B_1(\dot{x}_1 - \dot{z}_0) + F_1 + K_2(x_2 - x_1) + B_2(\dot{x}_2 - \dot{x}_1) - F_b - F_2 \quad (1)$$

$$M_2\ddot{x}_2 = -K_2(x_2 - x_1) - B_2(\dot{x}_2 - \dot{x}_1) + F_2 + K_3(x_3 - x_2) + B_3(\dot{x}_3 - \dot{x}_2) - F_3 \quad (2)$$

$$M_3\ddot{x}_3 = -K_3(x_3 - x_2) - B_3(\dot{x}_3 - \dot{x}_2) + F_3 \quad (3)$$

where the buffer force F_b is:

$F_b = K_b(x_1 - z - cl)$ when $x_1 - z - cl > 0$

$F_b = K_b(x_1 - z + cl)$ when $x_1 - z + cl < 0$

$F_b = 0$ otherwise

The buffer force is an additional force introduced to resist larger tremors by using a higher stiffness at greater deflections. The system can be soft for small tremors and stiff for larger ones.

The previous equations can be rewritten as:

$$M_1\ddot{x}_1 = -M_2\ddot{x}_2 - M_3\ddot{x}_3 - K_1(x_1 - z_0) - B_1(\dot{x}_1 - \dot{z}_0) - F_b + F_1 \quad (4)$$

$$M_2\ddot{x}_2 = -M_3\ddot{x}_3 - K_2(x_2 - x_1) - B_2(\dot{x}_2 - \dot{x}_1) + F_2 \quad (5)$$

$$M_3\ddot{x}_3 = -K_3(x_3 - x_2) - B_3(\dot{x}_3 - \dot{x}_2) + F_3 \quad (6)$$

Stiffness and viscous damping can be expressed as functions of natural frequencies and damping ratios as follows:

$\omega_j^2 = K_j / M_j$ (j =1,3)

and

$B_1 = 2\zeta_{soil}\ \omega_1 M_1$

$B_2 = 2\zeta_{str}\ \omega_2 M_2$

$B_3 = 2\zeta_{str}\ \omega_3 M_3$

where ζ is the damping ratio.

The input (figure 2) was obtained from a recorded ground input. [Guiclea and Sireteanu]

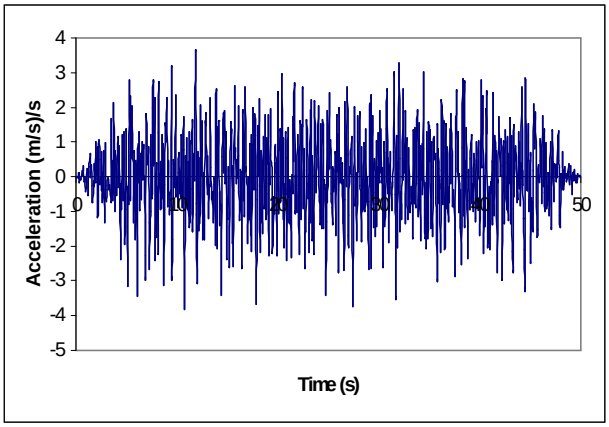


Figure 2. Time trace of earthquake

The simulated earthquake time history is Kcompatible with a design response spectrum used for testing an emergency diesel electric power generator from the heavy water plant of Turnu Severin, a town in the southern part of Romania.

The simulated quake is typical of the Romanian seismogenic zone, which is known to be different from other zones in other part of the world (e.g. California, Turkey, or Japan), in terms of range of accelerations and frequencies. Hence to fully test the performance of the semi-active system, additional tests might be required with different types of earthquakes from different seismogenic zones. The behaviour of MR fluids is here represented as a Bingham plastic model having variable yield strength. The schematic of this model is sketched in figure 3.

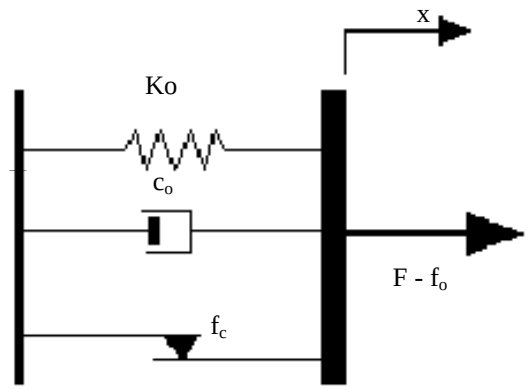


Figure 3. Bingham plastic model of the MR damper

The authors measured the force generated by a pilot 500 Newton MR damper during periodic motion at frequencies in the range 0.5 - 5 Hz in order to identify parameters. The current supplied was varied from 0.1 A to 1.75 A (figure 4 shows a damping characteristic). The issue of scalability of the damper has been investigated by Spencer *et al.* [1998]. They proved that with appropriate design the power required to control a 200 kN damper is only 50 W, hence the current required is of the order of the few amps**.** This allowed the damper characteristics to be derived experimentally using a Bingham model.

The damping force-displacement-velocity characteristic for a variable magnetic field is described by the following equation:

$$F(x, \dot{x}, u) = f_c(u)\, sgn(\dot{x}) + c_0(u)\, \dot{x} + K_0(u)\, x + f_0 \quad (7)$$

where c_0 is the damping coefficient, f_c the friction force directly related to the yield stress, K_0 the elastic coefficient (all current-dependent), and f_0 the offset force due to the compressed gas within the damper which prevents the damper collapsing when subject to a static load. x is the imposed relative displacement and u is the control current. Voltage-current dynamics in the solenoid are modelled via a first order lag with time constant of 10 ms.

More complicated models have been investigated by the authors, such as the Bouc-Wen model [Sireteanu *et al.*, 2001], which represents better the hysteretic behaviour in the pre-yield low velocity range. However for the purpose of this study a Bingham model has been used which, inter alia, also reduces computational load.

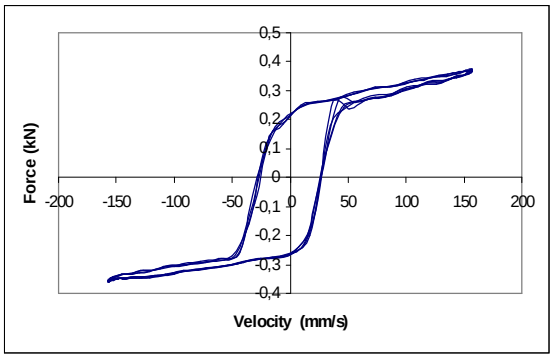


Figure 4. MR damper characteristics

4. Controller Design

The controller is designed using a variable structure control approach [Utkin, 1992]. With this approach the controller (and hence the "structure" of the system) switches between different laws according to a switching condition designed in order to obtain stability.
Three cases are considered

(a)
Base sliding only - the common passive isolation policy. F_s is set to a very low value so that spring 1 is not loaded. The clearance then allows small unrestrained motion of the building. This is a well proven solution for modest earthquakes, but in the case of larger earthquakes ground lateral motion will exceed any practical clearance.

(b)
Base sliding with adaptive damping at H/3 and 2H/3 using MR dampers.

(c)
Adaptive damping as in the previous case, but base sliding is only allowed until the buffer spring is contacted. Spring cancellation at the base is then introduced, the sliding element in the base necessarily being locked.

Spring cancellation (termed balance logic by the authors) has been shown to be effective [Stammers and Sireteanu, 1998 2000a, 2000b] in reducing lateral accelerations and drift. Pseudo-viscous damping can be introduced between each block using a MR device. The device always dissipates energy in this role and is therefore always on when damping is demanded. Moreover the inter-storey damper reduces drift when this exceeds some specified "trigger" value. The aim is to reduce drift to below 0.5% h (where h is inter-floor height) while keeping lateral accelerations below 2 m/s^2**.** (2000 NEHRP Recommended Provisions For New Buildings and Other Structures (FEMA 368)).

The adaptive damping strategy used for controlling the MR damper is as follows. When drift is low, the objective is to reduce lateral accelerations, which is helped by low damping. If drift becomes larger, increased damping is desirable to prevent unacceptable levels of drift being reached. It is necessary to generate an appropriate controlled MR damper force-velocity characteristic by opportunely controlling the solenoid current. The adaptive (in the sense explained above) control force can be expressed by the following functional equation:

$$F_j(x,\dot{x},u) = F_j(x,\dot{x},u_j(x,\dot{x})) \quad (8)$$

where

$$u_j(x,\dot{x}) = c_{jk} \mid x_k - x_j \mid \quad (9)$$

where j=1, k=2 for the damper between the base and the middle block and j=2, k=3 for the damper between the middle block and the top block, and c_{12} and c_{23} are damping constants.

The balance strategy aims at eliminating or reducing component acceleration by suitable choice of F_j [see eqns (4)-(6)]. The strategy is to demand control forces which tend to reduce acceleration when the following **"**dissipative" condition is met:

$$F_j \dot{x}_{rj} < 0 \quad (10)$$

where $\dot{x}_{rj}$ is the relative velocity across damper j.

In order to achieve acceleration minimisation, the demanded forces are expressed by:

$$F_1 = \gamma\,[M_2\ddot{x}_2 + M_3\ddot{x}_3 + K_1(x_1 - z_0) + B_1(\dot{x}_1 - \dot{z}_0) + F_b] \quad (11)$$

$$F_2 = \beta\,[M_3\ddot{x}_3 + K_2(x_2 - x_1) + B_2(\dot{x}_2 - \dot{x}_1)] \quad (12)$$

$$F_3 = \alpha[K_3(x_3 - x_2) + B_3(\dot{x}_3 - \dot{x}_2)] \quad (13)$$

where α, β, γ are in the range [0,1]. As the system is semi-active, the damper is only switched on when condition (10) is fulfilled; otherwise zero current is supplied.

5. Simulation Results

A two dimensional representation of a 12 storey building of total height H was considered with the first three natural periods of 1.32 s, 0.47 s and 0.33 s. This yields the following values for the natural angular frequencies $\omega_1 = 3.4\pi$ rad/s, $\omega_2 = \omega_3 = 1.6\pi$ rad/s. K_b was taken to be equal to K_1.

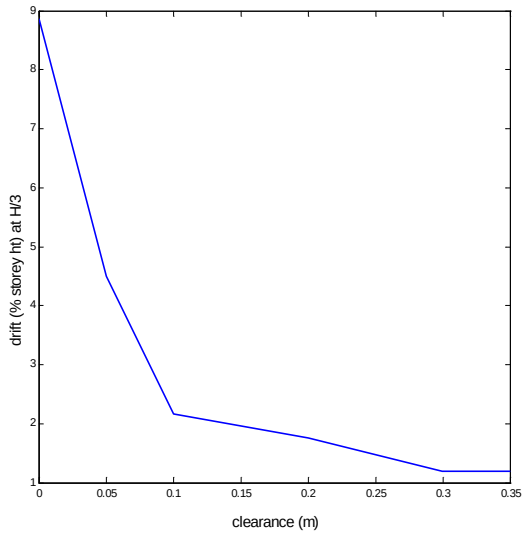


Figure 5. Drift at H/3, passive case

The response of the uncontrolled building to the earthquake considered [Guiclea and Sireteanu], scaled to a peak ground lateral acceleration of 6.68 ms^{-2} , was calculated for three equally sized blocks. The drift at one third height (which is greater than that at two-thirds height) is shown in figure 5 as a function of clearance. With zero clearance the earthquake is destructive, producing drift of over 8% storey height. Drift falls markedly with clearance, reaching a value of 1.2% with a clearance of 0.35m. However, as the maximum relative ground movement does not exceed 0.35m in this case, further increase of the clearance has no effect.

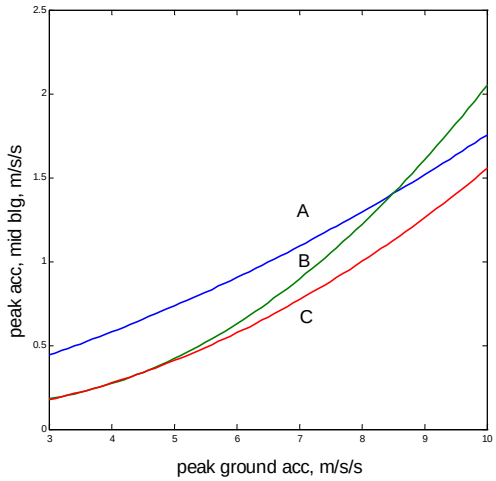


Figure 6. Drift at H/3 (A) sliding only (B) sliding with adaptive damping (C) sliding with adaptive damping and spring cancellation

The effect of MR control was now studied, with the earthquake input varied by scaling so that the peak input acceleration increased from 3.3 ms^{-2} to 9.81ms^{-2}. A clearance of 0.2 m was selected to ensure that impacts would occur unless control was applied.
Drift at H/3 in the control case is shown in shown in Fig 6 for three strategies A,B,C

For the adaptive law - see equation (8) and (9) - c_{12} and c_{23} were taken to be equal. Spring cancellation was applied at the base only. Previous work by the authors [Stammers and Sireteanu, 2000b] indicated that spring cancellation was more effective at the base than higher up the building. Referring to equations (11)-(13), the values adopted were $\alpha = \beta = 0$, $\gamma = 1$.
Pure sliding is inadequate (drift greater than 1%h) over the entire range of inputs. Adaptive damping ($c_{12} = c_{23}$=35) reduces drift substantially, while at 1g peak ground acceleration, the inclusion of spring cancellation achieves an 75% reduction compared to a sliding base alone.

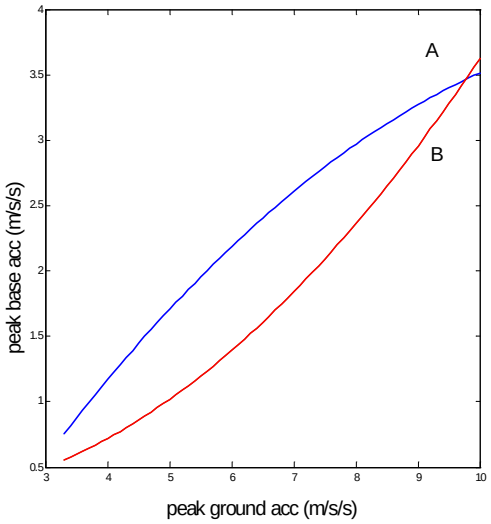


Figure 7. Base response

The peak acceleration of the base is plotted in figure 7 for systems A and B. Adding spring cancellation - system C - reduced response of the building in general, but not the accelerations at the base in this case and so is not plotted, as it is the same as for system B.
While adaptive damping is distinctly beneficial at low ground inputs, it is of no help at 1g inputs. The increased damping prevents excessive drift but conversely increases the force experienced by the base.

6. Issues on Control Implementation

The authors have developed this work starting from their previous work on friction dampers. The two devices although they are quite different in their physical principle, exhibit relatively similar characteristics. MR control logic described above could be also implemented with little modifications to the algorithm by means of friction dampers. The static characteristic of an experimental friction damper is similar to that of an MR damper. Bingham and Bouc-Wen models could be developed analogously. This allows the control logic to be implemented on either of them with relatively similar performance. However MR dampers offer substantial benefit in terms of physical implementation: in a controlled damping system employed for earthquake protection, the device may stay inactive for long periods and operate only in the event of an earthquake. Reliability is therefore a crucial issue in the design of any actuation system. Swift dynamic response is another requirement, since the device must be able to switch on rapidly in the occurrence of an earthquake and its bandwidth must be comparable with the dynamics of the seismic input.
These requirements are met by MR dampers which require only a coil supplied by a battery which provide the control current: MR dampers offer the significant advantage of an extremely simple actuation and very fast dynamics at start-up (the electrical dynamics of the solenoid). Furthermore an MR damper is an inherently fail-safe device, since in case of electrical supply failure, it turns into a conventional viscous damper, with the limit performance discussed above (case B versus C).
Conversely a friction damper is a device which is composed of a plate fixed to a mass with a pad pressed against the plate. An external normal force is applied to a mass by the pad and consequently, because of the relative motion between the pad and the plate and of the presence of friction, a damping force is produced.

In a friction damper the actuation system is far more involved. A friction damper for a structural application is typically activated hydraulically or, provided that the response is swift enough, pneumatically. In a hydraulically-actuated scheme this translates to pressure control, which requires a hydraulic circuit with a control valve supplied by a pump, working at a set constant pressure. The more natural design entails the use of pressure control valves, albeit this configuration is not the most effective in terms of number of hydraulic components.

Another design solution, successfully employed in a mobile application to a vehicle [Stammers *et al.*, 2000], [Guglielmino, 2001]) makes use of a 3-way underlapped flow control valve working in pressure control mode, exploiting the inherent pressure vs. spool opening characteristic, saturation-shaped, of the valve. The latter configuration is advantageous in cases where is necessary to control more dampers at the same time. By using pressure control valves, in fact it would be necessary to install as many pumps and valves as the number of controlled dampers, whereas using flow control valves, it is possible to use only one pump which supplies a main line and to position the valves in parallel, hence reducing the number of hydraulic components and therefore the overall reliability of the system. However despite the hydraulic drive can be optimised in terms of circuit configuration, lay-out and hence dynamic response, the merits of MR dampers are still paramount from an implementation viewpoint.

7. Conclusions

MR dampers, requiring only a modest electrical current input, are convenient devices for providing semi-control forces to counteract lateral ground inputs. The low time constant (values of 8 - 25ms are quoted) means there are no phase lag problems for typical seismic inputs.

Sliding base isolation is effective, but only in the case of earthquakes of modest size. A control strategy employing adaptive damping and spring cancellation applied to a sliding base system predicts up to 80% reductions in drift although there may be a compromise with increased accelerations. A logical extension would be to adopt an adaptive strategy in regard to lateral accelerations also. If lateral accelerations exceed a "trigger" level and drift is low, damping can be reduced and the spring "cancellation" parameters α, β and γ increased to reduce lateral accelerations.

Although previous work by the authors indicated that spring cancellation at the base was more effective in reducing lateral accelerations than if applied higher up the building, it is proposed to calculate the effect of non zero α and β when base sliding is also used.

Acknowledgements

Three of the authors (CWS, TS and DS) are supported by a Royal Society European Joint Project and wish to acknowledge their gratitude to the Royal Society for the support which made this work possible. Many thanks are due to Sarah Fuge for her skilled assistance .

8. References

Agrawal, A., Kulkarni, P., Vieira, S. L. and Naganathan, N. G. [2001] "An overview of magneto- and electro-rheological fluids and their applications in fluid power systems," *International Journal of Fluid Power* Vol. 2, pp. 5-36.

Akbay, Z. and Aktan, H. M. [1991] "Actively regulated friction slip devices," *Proc. 6th Canadian Conference on Earthquake Engineering*, pp. 367-374.

Casciati, F., Faravelli, L. and Yao, T. [1997] "Control of nonlinear structure using the fuzzy control approach", *Nonlinear dynamics*, Vol 11, pp. 171-187.

Dowdell, J. and Cherry, S. [1994] "Semi-active friction dampers for seismic response control of structures," *Proc. 5th US National Conference on Earthquake Engineering*, pp. 819-828.

Dyke, S. J., Spencer, B. F., Sain, M. K. and Carlson, J. D. [1996] "Modelling and control of magnetorheological dampers for seismic response reduction", *Smart Materials and Structures*, Vol 5, pp. 565-575.

Feng, M. Q. [1993] "Application of hybrid sliding isolation system to buildings," *Journal of Engineering Mechanics*, 119 (10), pp. 2090-2108.

Feng, M. Q. [1994] "Innovations in earthquake isolation," *The Active Control of Vibration*, IUTAM Symposium, Bath, UK, pp. 307-314.

Field, R. V., Voulgaris, P.G. and Bergman, L.A. [1996] "Methods to compute probabilistic measures of robustness for structural systems", *J Vibration and Control*, Vol. 2 (4), pp. 447-464.

Giuclea, M. and Sireteanu, T. [2000] "Optimisation of seismic accelerograms by genetic algorithms," *Proc 2nd International ICSC Symposium on Engineering of Intelligent Systems (EIS 2000)*.

Guglielmino, E., Stammers, C. W. and Edge, K. A. [2000] "Robust force control in electrohydraulic friction damper systems using a variable structure scheme with non-linear state feedback," *2nd Internationales Fluidtechnisches Kolloquium*, Dresden, Germany, pp. 163-176.

Guglielmino, E. [2001] "*Robust control of a hydraulically actuated friction damper for vehicle applications*," Ph.D. thesis, University of Bath, UK.

Nishitani, A., Nitta, Y. and Ishibashi, Y. [2000] "Structural control based on semi-active variable friction dampers," *12th World Conference on Earthquake Engineering*, Auckland, New Zealand, paper 1909 (CD-ROM).

Kobori, T., Takahashi, M., and Niwa, N. [1993] "Seismic response controlled structure with active variable stiffness," *Earthquake Engineering and Structural Dynamics*, Vol 22 (12), pp. 925-941.

Singh, M.P., Matheu, E. and Beattie, C. [1996] "Output feedback sliding mode control for civil engineering structures,". *Proc. 2nd Int Workshop on Structural Control*, Hong-.Kong, pp. 609-620.

Sireteanu, T., Stancioiu, D., and Stammers, C. W. [2001] "Modelling of magnetorheological fluid dampers", *Proc. Romanian Academy*, series A, 2, No 3/2001, pp. 105-113.

Spencer, B. F., Dyke, S.J., Sain, M. K. and Carlson, J. D. [1997] "Phenomenological model for magnetorheological dampers", *Journal of Engineering Mechanics*, Vol 123, pp 230-238.

Spencer, B. F. and Sain, M. K. [1998] "Controlling buildings: a new frontier in feedback," *Shock and Vibration Digest*, Vol 30, (4), pp. 267-281.

Stammers, C. W., Guglielmino, E. and Sireteanu, T. [1999] "A semi-active friction system to reduce machine vibration," *10th World Congress on the Theory of Machines and Mechanisms*, Oulu, Finland, Vol. 5, pp. 2168-2173.

Stammers, C. W. and Sireteanu, T. [1998] "Semi-active control of ground induced building vibrations using friction dampers", *MOVIC Conference*, Zurich, Switzerland, Vol. 1, pp. 51-54.

Stammers, C. W. and Sireteanu, T. [2000] "Dual semi-active friction dampers to reduce building response to seismic inputs," *12th World Conference on Earthquake Engineering*, Auckland, New Zealand, paper 2561 (CD-ROM).

Stammers, C. W. and Sireteanu, T. [2000] "Control of building seismic response by means of three semi-active friction dampers", *Journal of Sound and Vibration*, 237 (5), pp 745-759.

Tsopelas, P., Constantinou, M. C, Kim, Y. S. and Okamoto. S. [1996] "Experimental study of FPS system in bridge seismic isolation," *Earthquake Engineering and Structural Dynamics*, Vol 25, pp. 65-78.

Utkin, V. [1992] "*Sliding modes in control and optimization*", Springer-Verlag.

Yamamoto, M. and Aizawa, S. [1994] "Control effects of active mass damper system installed on actual buildings," *Proc 1st World Conf. on Structural Control*, Los Angeles, USA, FP1 pp. 13-22.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

Rigid- and Soft-body Simulation

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

SIMULATION OF LIQUID-, DEFORMABLE- AND RIGID-BODIES

Niels Boldt
boldt@diku.dk
University of Copenhagen
Department of Computer Science
Denmark

Abstract

Realistic interaction between rigid-, deformable- and liquid-bodies can add a substantial realism in interactive applications such as surgery simulators or computer games. In this paper we propose a particle based method to simulate the three type of bodies and their interactions. Compared to previous methods our method uses the same geometrical representation for all objects which makes collision detection and interaction particular simple. A simulator illustrating the capabilities of the proposed method has been implemented and results are shown.
Keywords: Simulation, Deformable-Bodies, Liquid-Bodies, Rigid-Bodies, Interaction, Particle Based

Nomenclature

$\mathbf{a}$	Acceleration
$\mathbf{a}_{cm}$	Acceleration of center-of-mass(CM)
c	Connection
C	Set of connections
e	Particle
E	A set of particles
$\mathbf{f}$	Force
$\mathbf{g}$	Gravity
$\mathbf{I}$	Inertia tensor
$\mathbf{I}_{body}$	Inertia tensor defined in body-space
k_c	Hooke spring constant.
l	Initial length of connection
$\mathbf{L}$	Angular momentum
m	Mass of a particle
M	Mass of a rigid-body
p	Pressure
$\mathbf{p}$	Position of a particle
r	Radius of particle
$\mathbf{r}_{cm}$	Position of CM for a rigid-body
$\mathbf{R}$	Orientation of a rigid-body
$\mathbf{v}$	Velocity of a particle
$\mathbf{v}_{cm}$	Velocity of CM
$\mathbf{W}$	Normalized integration kernel
ρ	Density
ω	Angular velocity
τ	Torque
μ	Viscosity

Introduction

Simulation of liquid-, deformable-, and rigid-bodies have found widespread use in games, surgery-simulators and virtual prototyping. Physical models to simulate the different bodies has been developed and applied with success in these areas. However, not much attention has been given to integration of the different physical models. This poses a problem for collision detection as the geometric representation often is different and interfaces between different physical models must be defined.

In this paper we try to remedy this problem by proposing physical models for all three kinds of bodies that can interact through the same interface.

Our work is based on [JV03] where physical models for rigid- and deformable-bodies are presented together with an interface between the physical models. A volumetric mass-spring model is used to simulate deformable-bodies and all interaction are handled between individual particles of the bodies.

We have extended their work with a physical model for liquid-bodies that can be fitted into the existing framework.

The rest of this paper is organized as follows: First we highlight previous work on hybrid simulation in *Previous Work*. In *Physical Models* the physical model is presented and in *Dynamics* dynamics of the bodies are presented. Our implementation and re-

sults is presented in *Implementation and Results* and we conclude the paper with a discussion of future work in *Conclusion and Future Work*.

Previous Work

In [BW97] a hybrid simulation method is proposed. A interleaved simulation paradigm is described where simulation results from the previous time step are feed to the other model in the current time step to produce synchronized motion. Rigid-bodies, cloth, and particle systems can be handled. In [JV03] physical models for rigid- and deformable-bodies are presented that allows for interaction between different bodies. Interactions in the form of forces are computed between individual particles and then applied internally in the physical model to produce synchronized motion. Wagenaar [Wag01] have developed a particle based method to simulate rigid- and deformable-bodies. Interactions are handled by creating constraints between the particles which in turn satisfied by relaxation. Force- and distance-based constraints are discussed. In [MM04] a method is presented to model interactions between liquid-particles and soft-tissue. Pseudo particles are placed onto the surface representing the tissue and the pseudo particles interact with the liquid particles via a Lennard-Jones potential. In [MCT04] a method is presented to model interactions between rigid-bodies and fluids. This is done by constraining the fluid inside the rigid-body to rigid-body motion.

Physical Models

In this section we present the physical model used to model the bodies. For simplicity we use forces to model interactions but other possibilities exists.

Definition 1. Particle: *A particle* e *is a set of parameters* $\{\mathbf{p}\ ,\ \mathbf{v}\ ,\ m\ ,\ r\}$ *where*

$\mathbf{p}$	*Position*
$\mathbf{v}$	*Velocity*
m	*Mass*
r	*Radius*

Particles part of a liquid-body are also assigned a density ρ and a pressure p which are used in the equations governing liquid motion.

Definition 2. Connection: *A connection* c *is a set of parameters* $\{e_1\ ,\ e_2\ ,\ k\ ,\ l\}$ *where*

$e_1\ ,\ e_2$	*Particles comprising the connection*
k	*Hooke spring constant*
l	*Nominal distance*

Connections are used to model both internal and external interactions. A connection can either be static or dynamic. Static connections are used to model interaction between particles in deformable-bodies. Dynamic connections are used to handle interactions between bodies or self-intersections in deformable-bodies. A static connection exist during a whole simulation while a dynamic connection are created when a pair of particles are within a certain distance threshold and destroyed again when the distance between the pair of particles are larger than the threshold.
The set of connections for a body will only consist of connections that can be created with particles from that body. All connections between different bodies will belong to an interface that handles interactions between the bodies.

Definition 3. Deformable-body: *A deformable-body is a set of parameters* $\{E\ ,\ C\}$

E	*Set of particles comprising the body*
C	*Set of connections*

A deformable-body can be seen as a set of particles connected by springs. The set C consist of static and dynamic connections. The static connections decides the elastic properties while the dynamic connections handles self-intersections inside the body. This model is comparable to other particle models [Pro95] except that a volume is assigned to each particle.

Definition 4. Rigid-body: *A rigid-body is a set of parameters* $\{\mathbf{r}_{cm}\ ,\mathbf{R}\ ,\ \mathbf{v}_{cm}\ ,\ \omega\ ,\ M\ ,\ \mathbf{I}_{body}\ ,\ E\}$

$\mathbf{r}_{cm}$	*Position of center of mass(CM)*
$\mathbf{R}$	*Orientation of the body*
$\mathbf{v}_{cm}$	*Velocity of* $\mathbf{r}_{cm}$
ω	*Angular velocity*
M	*Mass of the body*
$\mathbf{I}_{body}$	*Inertia tensor*
E	*Set of particles comprising the body*

A rigid-body is a deformable-body, except that the distances of the connections are fixed. A rigid body is treated as a single entity when determining its motion and therefore additional parameters are introduced to ease this treatment. However, these parameters exists implicitly in the particles, and we could

control the motion of a rigid-body by satisfying the fixed-length constraints in each time step. No connections should be created inside the body, as distances between rigid-body particles are fixed.

Definition 5. Liquid-Body: *A liquid-body is a set of parameters* $\{E\}$ *where*
E *Set of particles comprising the liquid-body*

A liquid-body can be viewed as a deformable-body, however, without static connections. Instead interactions between particles in the body will be handled by smoothed particle hydrodynamics [MCG03]. This will be treated in detail in the next section where we discuss dynamics of the bodies.

Definition 6. Interface: *An interface is a set of parameters* $\{C\}$ *where*
C *Set of connections.*

An interface is used to model interactions between bodies. Particles are connected when they are within a certain distance of each other. This resembles penalty-methods [MW88], where springs are inserted between colliding bodies, hence some of the weaknesses of the penalty method is transfered to our approach. Stiff differential equations and penetrations can occur if the parameters of a connection are not chosen with care. An advantage, is that insertion and removal of springs becomes easier, as it is based purely on a distance threshold between two particles.

Dynamics

In this section we will describe the mechanics governing motion of the bodies. For deformable- and liquid-bodies the motion is decided by the motion of an individual particle while rigid-bodies are treated as a single entity.

Deformable-Body Mechanics

The mechanics of a deformable-body is determined by the mechanics of each of its particles $e \in E$. For an individual particle we have according to Newton's second law

$$\mathbf{f} = m\mathbf{a} \tag{1}$$

The force is the sum of all forces acting on the particle. The force model from [JV02] is adapted in a simplified version as follows:

$$\mathbf{f} = \mathbf{f}_C + \mathbf{f}_G \tag{2}$$

where $\mathbf{f}_C$ is the force from the connections of the particle and $\mathbf{f}_G$ is a gravitational vector. Then we can write

$$\mathbf{f}_C = \sum_{i \in C_e} -k_c(\|\mathbf{p}_e - \mathbf{p}_i\| - l_c)\frac{\mathbf{p}_e - \mathbf{p}_i}{\|\mathbf{p}_e - \mathbf{p}_i\|} \tag{3}$$

and

$$\mathbf{f}_G = m\mathbf{g} \tag{4}$$

where $\mathbf{C}_e$ is the set of connections for the particle e, $\mathbf{p}_e$ is the position of the particle, k_c is a Hooke spring constant, l_c is initial length of the spring and m is the mass of the particle.

Rigid-Body Mechanics

For a rigid body we describe mechanics of center of mass (CM) and orientation. We will use $\mathbf{f}_e$ to denote the resulting force on each particle constituting the rigid-body. The resulting force on a rigid-body is computed by summing all contributions from the individual particles. The motion of CM is governed by Newton's second law

$$\mathbf{f}_{res} = \sum_{e \in E} \mathbf{f}_e = M\mathbf{a}_{cm} \tag{5}$$

The orientation is governed by the following equations

$$\tau = \frac{d\mathbf{L}}{dt} = \sum_{e \in E} (\mathbf{p}_e - \mathbf{r}_{cm}) \times \mathbf{f}_e \tag{6}$$

and

$$\omega = \mathbf{I}^{-1}L \tag{7}$$

where $\mathbf{I}$ is the inertia tensor in world space computed by $\mathbf{R}\mathbf{I}_{body}\mathbf{R}^T$, $\mathbf{L}$ is the angular momentum, and ω is the angular velocity. Finally the change in orientation can be written

$$\frac{d\mathbf{R}}{dt} = \tilde{\omega}\mathbf{R} \tag{8}$$

where $\tilde{\omega}$ is a cross product matrix

Liquid-Body Mechanics

To model the dynamics of a liquid-body we will use a interpolation method denoted *Smoothed Particle Dynamics* (SPH). SPH was invented to simulate nonaxisymmetric phenomena in astrophysics

and presented by Monaghan in [Mon92]. Since then, SPH has evolved to other fields including liquid simulation. Interactive simulation is obtained in [MCG03] which will form the basis for the dynamics we present in the following.

SPH is an interpolation method for particle systems. With SPH, field quantities that are only defined at discrete locations can be evaluated anywhere in space. According to SPH, a scalar quantity A is interpolated at location $\mathbf{p}$ by a weighted sum of contributions from all particles:

$$A_s(\mathbf{p}) = \sum_j m_j \frac{A_j}{\rho_j} \mathbf{W}(\mathbf{p} - \mathbf{p}_j, h) \tag{9}$$

where j iterates over all particles, A_j is the field quantity of $\mathbf{p}_j$ and ρ_j is the density. We will refer to this equation as the *SPH rule*. The function $\mathbf{W}(\mathbf{p}, h)$ is called the smoothing kernel with smoothing distance h.

The particle mass and density appear in Equation 9 because each particle represents a certain volume $V_j = m_j/\rho_j$. The mass is constant, but the density ρ_j varies and needs to be evaluated at each time step. Through substitution into Equation 9 we get for the density at location $\mathbf{p}$:

$$\begin{aligned} \rho_s(\mathbf{p}) &= \textstyle\sum_j m_j \frac{\rho_j}{\rho_j} \mathbf{W}(\mathbf{p} - \mathbf{p}_j, h) \\ &= \textstyle\sum_j m_j \mathbf{W}(\mathbf{p} - \mathbf{p}_j, h) \end{aligned} \tag{10}$$

Derivatives of field quantities can be obtained by ordinary differentiation.

When using SPH to derive fluid equations for particles care must be taken to ensure symmetric equations such that conservation of momentum and symmetry of forces are obtained.

In an Eulerian formulation, isothermal fluids are described by a velocity field $\mathbf{v}$, a density field ρ and a pressure field p. The evolution of these quantities over time are governed by two equations. The equation of continuity assures conservation of mass

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \tag{11}$$

and the equation of motion assures conservation of momentum

$$\rho \left(\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla)\mathbf{v} \right) = -\nabla p + \rho \mathbf{g} + \mu \nabla^2 \mathbf{v} \tag{12}$$

With particles, the conservation of mass is guaranteed and the equation of continuity can be omitted in favour of Equation 10.

In Equation 12 the term $\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla)\mathbf{v}$ can be replaced by the substantial derivative $D\mathbf{v}/Dt$. Thus the acceleration of a particle i can be written

$$\mathbf{a}_i = \frac{d\mathbf{v}_i}{dt} = \frac{\mathbf{f}_i}{\rho_i} \tag{13}$$

See [MCG03] for further details.

Application of the SPH rule to the pressure term in the equation of motion and symmetrizing yields

$$\mathbf{f}_i^{pressure} = \sum_b m_j \frac{\rho_j + \rho_i}{2\rho_j} \nabla \mathbf{W}(\mathbf{p}_i - \mathbf{p}_j, h)$$

Application of the SPH rule to the viscosity term in the equation of motion and symmetrizing yields

$$\mathbf{f}_i^{viscosity} = \mu \sum_j m_j \frac{\mathbf{v}_j - \mathbf{v}_i}{\rho_j} \nabla^2 \mathbf{W}(\mathbf{p}_i - \mathbf{p}_j, h)$$

See [MCG03] for further details. In [MCG03] surface tension is modelled using a method proposed in [Mor00]. To relate density to pressure a equation of state is needed. We adopt a variant of the ideal gas law

$$p = k(\rho - \rho_0) \tag{14}$$

as suggested by Desbrun [DC96]

Kernels

Stability, accuracy and speed of the SPH method highly depend on the choice of smoothing kernels. In [MCG03] the following kernels was designed with the above mentioned criterion's in mind.

The following kernel

$$\mathbf{W}(\mathbf{p}, h) = \frac{315}{64\pi h^9} \begin{cases} (h^2 - |\mathbf{p}|^2)^3 & 0 \le |\mathbf{p}| \le h \\ 0 \end{cases}$$

is used in all but two cases. An important feature of this kernel is that $\mathbf{p}$ only appears squared which means that it can be evaluated without the use of square roots in distance computations. However, if this kernel is used for computation of pressure forces particles tend to build clusters under high pressure. This happens because the gradient approaches zero near the origin. Desbrun [DC96] solves this problem

by using a kernel with non-vanishing gradient near the origin. Desbrun's kernel is given by

$$\mathbf{W}(\mathbf{r},h) = \frac{15}{\pi h^6} \begin{cases} (h-|\mathbf{p}|)^3 & 0 \le \mathbf{p} \le h \\ 0 \end{cases}$$

For computation of viscosity forces a third kernel

$$\mathbf{W}(\mathbf{p},h) = k \begin{cases} -\frac{|\mathbf{p}|^3}{2h^3} + \frac{|\mathbf{p}|^2}{h^2} + \frac{2h}{|\mathbf{p}|} - 1 & 0 \le |\mathbf{p}| \le h \\ 0 \end{cases}$$

was designed where $k = \frac{15}{2\pi h^3}$. This kernel has the property that it always will decrease the fluids kinetic energy. See [MCG03] for further details. It is noted, that the use of this kernel for viscosity computations significantly increased the stability of the liquid simulation.

Implementation and Results

A interactive simulator implementing the described method has been constructed.

Since the smoothing kernels used in SPH have finite support h, the computational complexity has been reduced by using a grid with cells of size h. Thereby potentially interacting partners for a particle i only need to be searched in i's own cell and the neighbours. This technique reduces the computational complexity from $\mathcal{O}(n^2)$ to $\mathcal{O}(nm)$ where m is the average number of particles per grid cell.

This grid has also been used to accelerate collision queries between the bodies, as all particles of the bodies are mapped into the grid.

Integration has been tested with a fourth order Runge-Kutta method, Heun predictor-corrector and the simple Euler step [Hea02]. The Heun predictor-corrector was found to be best with respect to computation time and stability.

In Figure 1 a series of pictures from two simulations of rigid- and liquid-bodies are shown. In Figure 2 a series of pictures from two simulations of rigid- and deformable-bodies are shown. The simulations shows the capabilities of the implemented simulator to handle different body types in the same configuration. Movies can be obtained from *www.boldts.dk/thesis/test* where other test also are presented.

The liquid is visualized by drawing the particles as spheres. This is sufficient to illustrate the results but for proper use better techniques for visualization should be applied.

Conclusion and Future Work

We have presented a particle-based method to unify simulation of rigid-, deformable and liquid-bodies. Currently the implemented simulator can be termed interactive.

Collision queries with rigid- and deformable-bodies should be handled with a bounding volume hierarchy as proposed by Larsson [LAM01] and Gotschalk [GLM96]. Faster integration schemes could be implemented. A global adaptive scheme as proposed in [DC96] based on the Courant-Friedrichs-Lewy (CFL) criterion or a local adaptive scheme as proposed in [HK89] also based on the CFL criterion. With these optimizations simulation where the computational time is less than simulated time appears feasible. This is also known as real-time simulation.

Methods to visualize the liquids must be investigated. Currently no interactive method matching results of offline rendering exists. In [MCG03] point splattering and marching cubes are proposed. It is however concluded that more research must be done both to improve quality and performance.

In the future an GPU based approach would be interesting to investigate. The grid accelerating the collision queries could be replaced by a 3D-texture and pixel shaders should be used to compute interactions between particles. This should give a significant performance boost and release the CPU for other tasks.

An interface based on projections is being researched and promising results have been obtained. This however demands a change of state functions for the bodies. An interface where interaction between bodies are handled by SPH equations could be a possible direction for further research. The idea is presented in [SK98].

Acknowledgements

The author would like to thank Kenny Erleben and Jon Sporring for fruitful discussions and thorough proofreading.

References

[BW97] David Baraff and Andrew Witkin. Partitioned dynamics. Technical report, The Robotics Institute, Carnegie Mellon University, march 1997.

[DC96] Mathieu Desbrun and Marie-Paule Cani. Smoothed particles: A new paradigm for animating highly deformable bodies. In R. Boulic and G. Hegron, editors, *Computer Animation and Simulation '96 (Proceedings of EG Workshop on Animation and Simulation)*, pages 61–76. Springer-Verlag, Aug 1996. Published under the name Marie-Paule Gascuel.

[GLM96] S. Gottschalk, M. C. Lin, and D. Manocha. OBBTree: A hierarchical structure for rapid interference detection. *Computer Graphics*, 30(Annual Conference Series):171–180, 1996.

[Hea02] Michael T. Heath. *Scientific Computing - An Introductory Survey*. McGraw-Hill, second edition, 2002.

[HK89] L. Hernquist and N. Katz. TREESPH - A unification of SPH with the hierarchical tree method. *The Astrophysical Journal Supplement series*, 70, june 1989.

[JV02] J. Jansson and J. S. M. Vergeest. A discrete mechanics model for deformable bodies. *Computer-Aided Design*, 34, 2002.

[JV03] Johan Jansson and Joris S.M. Vergeest. Combining deformable- and rigid-body mechanics simulation. *The Visual Computer*, 2003.

[LAM01] Thomas Larsson and Tomas Akenine-Möller. Collision detection for continuously deforming bodies. Eurographics, pages 325–333, 2001.

[MCG03] Matthias Müller, David Charypar, and Markus Gross. Particle-based fluid simulation for interactive applications. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 154–159. Eurographics Association, 2003.

[MCT04] Peter J. Mucha Mark Carlson and Greg Turk. Rigid fluid: Animating the interplay between rigid bodies and fluid. 2004.

[MM04] M. Teschner M. Mueller, S. Schirm. Interactive blood simulation for virtual surgery based on smoothed particle hydrodynamics. *Journal of Technology and Health Care*, 12(1), 2004.

[Mon92] J. J. Monaghan. Smoothed particle hydrodynamics. *Annual Review of Astronomy and Astrophysics*, 30(1):543–574, 1992.

[Mor00] Joseph P. Morris. Simulating surface tension with smoothed particle hydrodynamics. *International Journal for Numerical Methods in Fluids*, 33, 2000.

[MW88] Matthew Moore and Jane Wilhelms. Collision detection and response for computer animationr3. In *Proceedings of the 15th annual conference on Computer graphics and interactive techniques*, pages 289–298. ACM Press, 1988.

[Pro95] Xavier Provot. Deformation constraints in a mass-spring model to describe rigid cloth behavior. In Wayne A. Davis and Przemyslaw Prusinkiewicz, editors, *Graphics Interface '95*, pages 147–154. Canadian Human-Computer Communications Society, 1995.

[SK98] Yoshiaki Oka Seiichi Koshizuka, Atsushi Nobe. Numerical analysis of breaking waves using the moving particle semi-implicit method. *International Journal for Numerical Methods in Fluids*, 26:751 – 769, 1998.

[Wag01] Jeroen Wagenaar. *Physically based simulation and visualization - A particle based approach*. PhD thesis, The Mærsk Mc-Kinney Møller Institute for Production Technology, 2001.

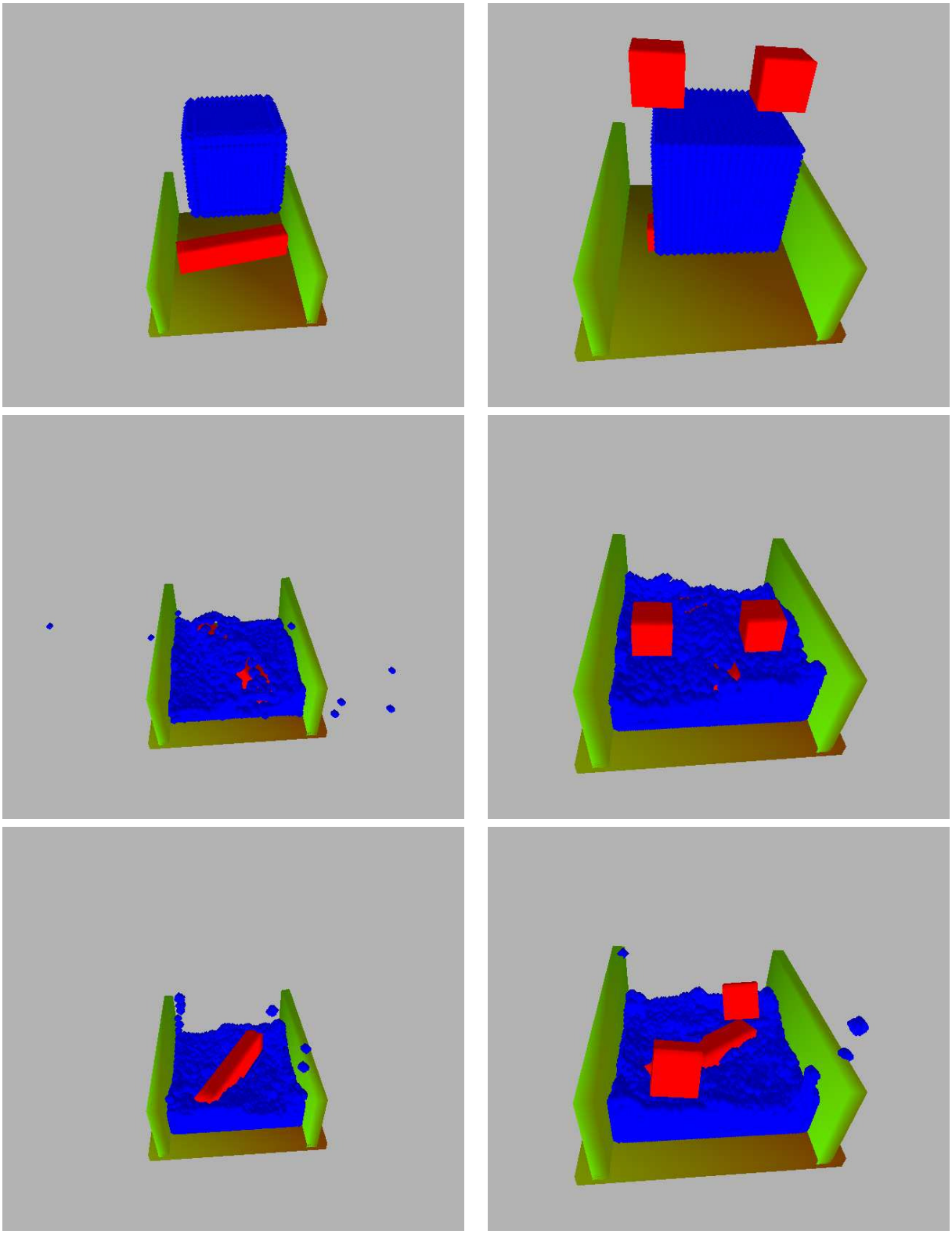


Figure 1: To the left, a sequence of frames from a simulation where a liquid-body is dropped into a vat with a moving box is shown. The box are scripted. To the right, a sequence of frames is shown from a simulation were a liquid-body is dropped into a vat together with two rigid-bodies. The vat contains a scripted box.

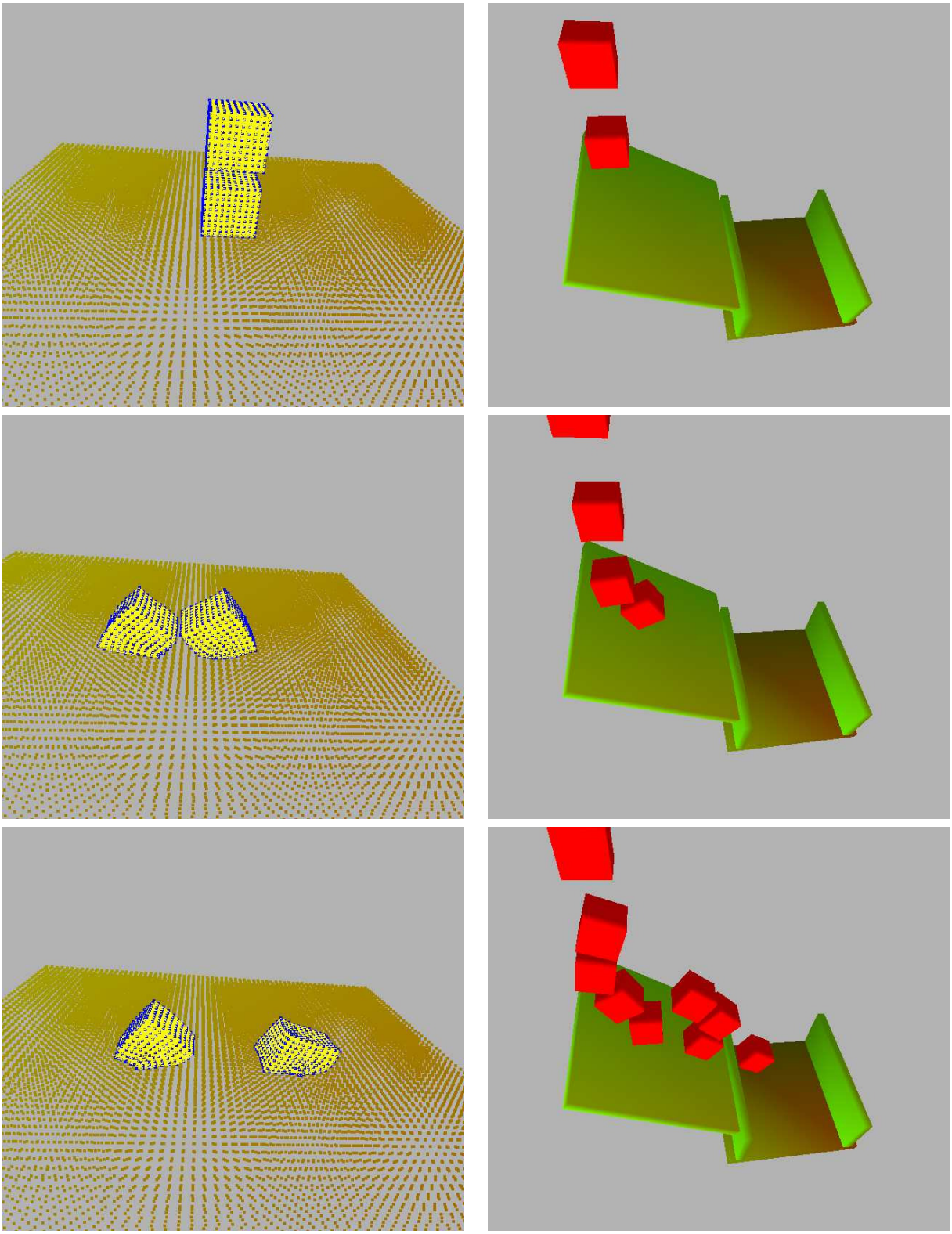


Figure 2: To the left, two deformable-bodies dropped on a moving plate are shown, all bodies are represented by particles drawn as spheres. The sequence of frames on the right side displays rigid-bodies being dropped into a vat.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

ANALYSIS OF MUSCULOSKELETAL SYSTEMS IN THE ANYBODY MODELING SYSTEM

Michael Damsgaard[1], John Rasmussen, Søren Tørholm Christensen, Egidijus Surma, and Mark de Zee
Institute of Mechanical Engineering, Aalborg University, 9220 Aalborg East, Denmark

Abstract

This paper introduces the simulation software the *AnyBody Modeling System*, the foundation of which was developed at Aalborg University by the authors. AnyBody is capable of analyzing the musculoskeletal system of humans or other creatures as rigid-body systems. The paper introduces the main features of the system, in particular, the inverse dynamic analysis that resolves the fundamental indeterminacy of the muscle configuration of musculoskeletal systems. The software was named AnyBody due to the ambition to make a modeling system by which the user can build musculoskeletal models of *any body*, or parts hereof. In addition to the musculoskeletal system, the models can comprise external objects, loads, and motion specification, thereby providing a complete set of the boundary conditions for a given task.
Keywords: Musculoskeletal models, muscles, multibody dynamics, inverse dynamics, simulation software.

Nomenclature

$\mathbf{C}^{()}$	Coefficient matrix of equilibrium equations.
$\mathbf{d}^{()}$	Right-hand side of equilibrium equations.
$\mathbf{f}^{()}$	Vector of forces. f_i is the i'th element.
$\mathbf{g}^{()}$	Vector of forces corresponding to **v**'s basis.
G	Objective functions.
$\mathbf{J}_i$	Inertia tensor of the i'th body.
$l_{i,}^{(oi)}$	Origin-insertion length of the i'th muscle.
m_i	The mass of the i'th body.
$n^{()}$	Integer number of e.g. bodies in the system.
N_i	Normalizing factor, typically muscle strength.
p	Polynomial degree.
$\mathbf{p}_i$	Vector of Euler parameters of the i'th body.
$\mathbf{q}$	Vector of position coordinates.
$\mathbf{q}^*$	Virtual positions corresponding to **v**.
$\mathbf{r}_i$	Translation vector of the i'th body.
$\mathbf{v}$	Vector of velocity coordinates.
γ	Right-hand side of acceleration constraints.
Φ	Vector of kinematic constraints violations.
$\Phi_{\mathbf{q}}$	Jacobian matrix with respect **q**.
ω_i	Angular velocity vector of the i'th body.

Quantities marked with $^{()}$ may have superscripts (M), (R) referring to 'muscles' and 'reactions', respectively.

Introduction

The musculoskeletal systems of humans or animals are mechanically very complex and computational models must be highly simplified in order to be reasonably efficient. Typically the musculoskeletal system is assumed to be a rigid-body system allowing for standard methods of multibody dynamics to be applied.

In addition however, the model must have reasonable representations of the muscle geometry and the recruitment pattern of the muscles, which are both complicated issues. The muscles consist of soft tissue and they wrap each other and the bones, ligaments, etc. in complicated shapes. Reasonable modeling of these geometries is essential for the mechanical model. Moreover, the muscles are activated by the Central Nervous System (CNS) by mechanisms that are not understood well enough for detailed modeling.

[1] Corresponding author: Phone: +45 9635 9310, Fax: +45 9815 1675 , E-mail: md@ime.aau.dk

Therefore, the modeling of these mechanisms is based on assumptions, typically some kind of optimality condition. The fundamental problem is that there are more muscles than necessary to drive the degrees of freedom of the system, which implies that there are infinitely many muscle recruitment patterns that are dynamically acceptable. This problem is often referred to as the redundancy problem of the muscle recruitment.

Musculoskeletal models can be divided into two groups: forward and inverse dynamic models. Forward dynamics computes the motion based on a predicted muscular activation. While this is attractive in the view of the detailed modeling of the physical phenomena, it is very computationally demanding and typically the model must be wrapped in a computationally costly optimization of the activation control to make the model do a specific task. Inverse dynamics computes the muscle activation based on a specified task, i.e., motion. This puts many restrictions on the model but the computations are much more efficient. This efficiency can be exploited to build more complex models comprising more muscles, i.e., more details of the real body.

The literature on biomechanics contains many such models. [1]-[3] show applications of forward dynamics, whereas many of the other references made later in this paper are for inverse dynamic models.

The AnyBody system, which was initiated at Aalborg University by the authors, is a general modeling system for making such musculoskeletal models. It was designed to meet four goals:

1) It should be a modeling system, i.e. a tool that allows users to construct models from scratch or use or modify existing models to suit different purposes.
2) The system should facilitate model exchange and allow models to be scrutinized.
3) If possible, it should have sufficient numerical efficiency to allow ergonomic design optimization on inexpensive computers.
4) The system should be capable of handling body models with a realistic level of complexity such as the example of Figure 1.

Currently, AnyBody allows only for inverse dynamic analysis of the models and therefore this paper will focus on this approach.

In the rest of this paper we shall attempt to give an introduction to the functionality of AnyBody, hereunder the theory behind the basic analysis capabilities and aspects of the software design.

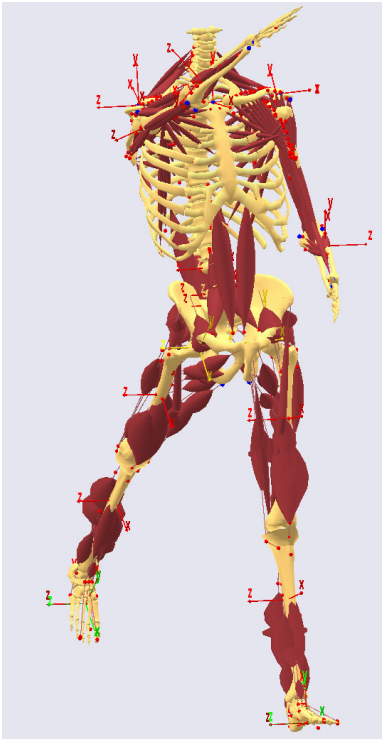


Figure 1: Full body model comprising several hundreds of muscles made with AnyBody.

Muscle Recruitment

The solution of the muscle recruitment problem in the inverse dynamics approach is generally formulated as an optimization problem on the form:

$$\underset{\mathbf{f}}{\text{Minimize}} \quad G(\mathbf{f}^{(\mathrm{M})}) \tag{1}$$

$$\text{Subject to} \quad \mathbf{C}\mathbf{f} = \mathbf{d}\,, \tag{2}$$

$$0 \le f_i^{(\mathrm{M})} \le N_i\,, \qquad i \in \left\{1,\ldots,n^{(\mathrm{M})}\right\} \tag{3}$$

where G is the objective function, i.e., the assumed criterion of the recruitment strategy of the CNS, stated in terms of the muscle forces, $\mathbf{f}^{(\mathrm{M})}$. G is minimized with respect to all unknown forces in the problem, $\mathbf{f}=[\mathbf{f}^{(\mathrm{M})\mathrm{T}}\ \mathbf{f}^{(\mathrm{R})\mathrm{T}}]^{\mathrm{T}}$, i.e., muscle forces, $\mathbf{f}^{(\mathrm{M})}$, and joint reactions, $\mathbf{f}^{(\mathrm{R})}$. Equation (2) is the dynamic equilibrium equations, which enter as constraints in the optimization. $\mathbf{C}$ is the coefficient-matrix for the unknown forces and the right-hand side, $\mathbf{d}$, contains all known applied loads and inertia forces. The non-negativity constraints on the muscle forces, (3), state that muscles can only pull, not push, and the upper bounds limit their capability, i.e., N_i is the strength of the muscle.

The most popular form of the objective function, G, is *the polynomial criteria* in (4) and a somewhat less known form, introduced by Siemienski [4], is *the soft saturation criteria* in (5).

$$G(\mathbf{f}^{(\mathrm{M})}) = \sum_{i=1}^{n^{(\mathrm{M})}} \left(\frac{f_i^{(\mathrm{M})}}{N_i} \right)^p \tag{4}$$

$$G(\mathbf{f}^{(\mathrm{M})}) = -\sum_{i=1}^{n^{(\mathrm{M})}} \sqrt[p]{\left(1 - \frac{f_i^{(\mathrm{M})}}{N_i} \right)^p} \tag{5}$$

Both of these forms of G are stated with a variable power, p, and a normalizing function for each muscle, N_i. The normalized muscle force is often referred to as the *muscle activity*.

The most physiologically reasonable choice of N_i is some measure of the strength of the muscle. This can either be approximated by a constant or computed by some model taking the muscle's operating conditions into account. The modified Hill model introduced by Zajac [5] is a popular choice using the length and the length-rate of the muscle as input. These will be available from the kinematical analysis of the system.

More details about these criteria can be found in the literature, e.g. [6]-[10] apply the polynomial criteria with various powers, p, and Siemienski [4] introduces the soft saturation criterion with p=2. Thorough comparison and discussion of different criteria are found in [11]-[16],

A third possibility is the so-called min/max formulation that takes the form:

$$G(\mathbf{f}^{(\mathrm{M})}) = \max\left(\frac{f_i^{(\mathrm{M})}}{N_i} \right), \tag{6}$$

i.e., minimization of the maximal muscle activity. A number of things make this criterion attractive compared to (4) and (5).

Firstly, it can be transformed into a linear problem, which makes it numerically efficient and possible to solve with a finite algorithm, see [15] and [16]. (4) and (5) also have this feature for p=1; however, it is generally agreed that p=1 leads to a physiologically unreasonable result, namely that the stronger muscles do all the work and real muscles are known to share the loads whenever possible.

For higher powers, p, (4) and (5) become less and less numerical attractive. Moreover, the polynomial criteria must be equipped with the upper bound constraints in (3) on the muscle activity, whereas both the soft saturation and the min/max criteria have these constraints implicitly fulfilled. The soft saturation criteria, however, cause numerical problems when activities are close to the upper limit, whereas the min/max criterion simply utilizes the muscles optimally so that the activities do not exceed the limit before absolutely unavoidable. Also this handling of the upper bound makes (6) numerically attractive.

Finally, one should notice that Rasmussen et. al., [14], showed that (4) and (5) converge towards each other for increasing power, p, and they do indeed converge towards (6). In addition, we notice that that the activity's dependency on the magnitude of external load converges towards a linear function for large p and it is indeed a linear relationship for the min/max criterion, see [14].

This convergence is an important result in the view that Challis and Kerwin in [11] compared the polynomial criteria with experiments and found the best agreement for very large powers (p=100). Numerically, this is equivalent to a solution obtained using (6).

Apart from being numerically attractive the criterion in (6) also appears to be physiologically attractive. Assuming that muscle fatigue and activity are proportional the criterion postpones fatigue as much as possible; in other words it is a minimum fatigue criterion.

Based on this line of argumentation the criterion (6) was chosen as the foundation of the inverse dynamics analysis in the AnyBody software.

It should however be emphasized that none of the presented criteria has been generally proven as the superior one physiologically.

Some believe that (6) leads to too much muscle synergism, [16], and indeed it does exploit the muscles with very poor working conditions, i.e., small moment arms, to a degree that may be questionable. Also a minor numerical difficulty of (6) should be mentioned. It inherently contains some indeterminacy for certain groups of sub-maximally activated muscles, which may be obvious since the objective only involves the maximally activated ones. This must be dealt with to obtain a unique solution out of (6) for complex models. [17] solves this by an additional criterion, whereas [15] and [16] suggest a dedicated iterative solution, which is implemented in AnyBody.

Mechanical Model

The mathematical model of the mechanical system must produce the equations of motion in the form of (2). We adopt a general multibody system dynamics approach using a set of Cartesian coordinates for each body. This is a choice of generality and ease of implementation over efficiency of the kinematical analysis. This is a reasonable choice because the kinematics is a minor part of the analysis, where we typically deal with many more muscles than segments considering for instance a full body model.

All segments of the biomechanical system are modeled as rigid bodies[2], neglecting effects such as the wobbly masses of soft tissues. We, more or less, adopt the formulation in the textbook by Nikravesh [18]. The position of the i'th body is described by the coordinates $\mathbf{q}_i=[\mathbf{r}_i^{\mathrm{T}}\ \mathbf{p}_i^{\mathrm{T}}]^{\mathrm{T}}$, where $\mathbf{r}_i$ is the global position vector of the center of mass and $\mathbf{p}_i$ is a vector of four Euler parameters. The velocity of the bodies is defined as $\mathbf{v}_i=[\dot{\mathbf{r}}_i^{\mathrm{T}}\ \omega_i'^{\mathrm{T}}]^{\mathrm{T}}$, where the vector ω_i' is the angular velocity of the body measured in the body-fixed reference frame.

The kinematical analysis is carried out in terms of all the Cartesian coordinates by solving a set of imposed kinematical constraints of the form

$$\Phi(\mathbf{q},t)=\mathbf{0}\,, \tag{7}$$

where $\mathbf{q}=[\mathbf{q}_1^{\mathrm{T}}\ ..\ \mathbf{q}_n^{\mathrm{T}}]^{\mathrm{T}}$ is the assembled coordinate vector for all n segments. The explicit time, t, indicates that some of the constraints are kinematical drivers in addition to the constraints arising from the joints. In the case of inverse dynamic analysis the imposed constraints must specify the motion completely, implying that we have a full set of equations in (7). Notice, we must also include the unity constraints on the Euler parameters. The equations in (7) are generally non-linear and we solve them by a modified Newton-Raphson scheme. Subsequently, we solve the linear velocity and acceleration constraints, (8), but in terms of $\mathbf{v}$ and $\dot{\mathbf{v}}$ instead of time-derivatives of $\mathbf{q}$:

$$\Phi_{\mathbf{q}^*}\mathbf{v}=-\Phi_t \quad \text{and} \quad \Phi_{\mathbf{q}^*}\dot{\mathbf{v}}=\gamma(\mathbf{q},\mathbf{v},t)\,, \tag{8}$$

where $\Phi_{\mathbf{q}^*}$ is the Jacobian with respect to $\mathbf{q}^*$. $\mathbf{q}^*$ contains a virtual set of positions that correspond to $\mathbf{v}$. They are not meaningful as finite values due to the rotational entries in $\mathbf{v}$, but as infinitesimal values in differentiations they make sense.

Now, we know the motion completely in $\mathbf{q}$, $\mathbf{v}$, and $\dot{\mathbf{v}}$. We can now turn towards the target of setting up the dynamics equilibrium, (2). For each segment, we have the Newton Euler equations, which in this case may take the form:

$$\begin{bmatrix} m_i\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{J}_i' \end{bmatrix}\dot{\mathbf{v}}_i+\begin{bmatrix} \mathbf{0} \\ \tilde{\omega}_i'\mathbf{J}_i'\omega_i' \end{bmatrix}=\mathbf{g}_i \tag{9}$$

where m_i and J_i' are the mass and the inertia tensor referring to the centroidal body-frame, respectively. The right-hand side, $\mathbf{g}_i$, is the forces, having six entries, firstly three forces and then three moments in body-fixed coordinates. It consists of muscle forces, $\mathbf{g}_i^{(\mathrm{M})}$, reaction forces, $\mathbf{g}_i^{(\mathrm{R})}$, and known applied loads, $\mathbf{g}_i^{(\mathrm{app})}$, that may depend on $\mathbf{q}$, $\mathbf{v}$, and t. $\mathbf{g}_i^{(\mathrm{M})}$ and $\mathbf{g}_i^{(\mathrm{R})}$ enter (2) on the left-hand side, whereas the remaining entries in (9) enter $\mathbf{d}_i$; thus the full right-hand side of (2) is assembled as $\mathbf{d}=[\mathbf{d}_1^{\mathrm{T}}\ ..\ \mathbf{d}_n^{\mathrm{T}}]^{\mathrm{T}}$, where:

$$\mathbf{d}_i=\mathbf{g}_i^{(\mathrm{app})}-\begin{bmatrix} m_i\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{J}_i' \end{bmatrix}\dot{\mathbf{v}}_i-\begin{bmatrix} \mathbf{0} \\ \tilde{\omega}_i'\mathbf{J}_i'\omega_i' \end{bmatrix} \tag{10}$$

Similar to the unknown forces, $\mathbf{f}=[\mathbf{f}^{(\mathrm{M})\mathrm{T}}\ \mathbf{f}^{(\mathrm{R})\mathrm{T}}]^{\mathrm{T}}$, their coefficient matrix, $\mathbf{C}$, can be partitioned according to muscle and reaction forces, i.e., $\mathbf{C}=[\mathbf{C}^{(\mathrm{M})}\,\mathbf{C}^{(\mathrm{R})}]$ that define $\mathbf{g}^{(\mathrm{M})}=\mathbf{C}^{(\mathrm{M})}\,\mathbf{f}^{(\mathrm{M})}$ and $\mathbf{g}^{(\mathrm{R})}=\mathbf{C}^{(\mathrm{R})}\,\mathbf{f}^{(\mathrm{R})}$.

$\mathbf{C}^{(\mathrm{R})}$ is in principle the transposed of the constraint Jacobian, $\Phi_{\mathbf{q}^*}$. This follows from standard forms of the constrained equations of motion where $\mathbf{f}^{(\mathrm{R})}$ is given by Lagrange multipliers of the constraints, see e.g. [18]. A constraint imposed on the motion by a mechanical device corresponds to a reaction component in that device, i.e., each row in $\Phi_{\mathbf{q}^*}$ corresponds to a column in $\mathbf{C}^{(\mathrm{R})}$. However the kinematical constraints in (7) and (8) also contain pure motion specification of the system's degrees of freedom. We shall exclude these constraints from $\mathbf{C}^{(\mathrm{R})}$ so that $\mathbf{C}^{(\mathrm{R})}$ becomes the subset of the columns from the transposed Jacobian, $\Phi_{\mathbf{q}^*}$, that are associated with the real mechanical devices. The neglected columns correspond to the system's degrees of freedom and these will be kinetically supported by muscles.

The muscle coefficient matrix, $\mathbf{C}^{(\mathrm{M})}$, requires a geometric model of the muscles. We model the muscles geometrically as elastic strings spanning

[2] In biomechanics we speak of segments instead of (rigid) bodies, because "body" can be confused with the human body.

between two or more points and in cases wrapping over rigid obstacles. It is beyond the scope to describe the wrapping model here, but in the simple case without wrapping, we can express the muscle's *origin-insertion length* as $l^{(oi)}= |\mathbf{r}_i^{(p)} - \mathbf{r}_j^{(p)}|$, where $\mathbf{r}_i^{(p)}$ and $\mathbf{r}_j^{(p)}$ are the positions of the spanned points that depend on $\mathbf{q}$. Any other model of the muscle path must similarly provide us with $l^{(oi)}(\mathbf{q})$ and its time-derivative for calculation of the strength, N_i. By the principle of virtual work, we can show that the coefficients in $\mathbf{C}^{(M)}$ are the derivatives of $l^{(oi)}$ with respect to the system coordinates in $\mathbf{q}^*$. These derivatives we shall denote this by $l_{i,\mathbf{q}^*}^{(oi)}$. Firstly we can express the virtual work produced by the muscles as the sum of muscle forces times their virtual length change:

$$\begin{aligned}\delta W &= \sum_{i=1}^{n^{(M)}} \delta l_{i,\mathbf{q}^*}^{(oi)} f_i^{(M)} = \delta\mathbf{q}^{*T} \sum_{i=1}^{n^{(M)}} l_{i,\mathbf{q}^*}^{(oi)} f_i^{(M)} \\ &= \delta\mathbf{q}^{*T} \left[l_{1,\mathbf{q}^*}^{(oi)} .. l_{n^{(M)},\mathbf{q}^*}^{(oi)} \right] \mathbf{f}^{(M)}\end{aligned} \tag{11}$$

Secondly, we can express the virtual work as the scalar product of the generalized force vector for all muscles, $\mathbf{g}^{(M)}$, and the virtual change of the system coordinates $\mathbf{q}^*$:

$$\delta W = \delta\mathbf{q}^{*T}\ \mathbf{g}^{(M)} = \delta\mathbf{q}^{*T}\ \mathbf{C}^{(M)}\mathbf{f}^{(M)} \tag{12}$$

Comparison of (11) and (12) shows that $l_{i,\mathbf{q}^*}^{(oi)}$ is indeed the i'th column of $\mathbf{C}^{(M)}$.

We have now established all entries of the muscle recruitment problem in (1)-(3). In order to speed up the actual solution, we can optionally reduce the number of equilibrium equations in (2). Since there are no other conditions on the reaction forces, $\mathbf{f}^{(R)}$, than equilibrium equations, we can apply a standard factorization to eliminate elements of $\mathbf{f}^{(R)}$ using an equal amount of equations of (2). We shall however not go into further details about this.

The AnyBody software

The AnyBody software consist of two applications, a Windows (GUI) and a console application, that both have the same modeling facilities but differ in the ways they can be used. The console application can be called from other programs whereas the GUI application contains much more facilities for viewing the model and its results, thereby providing a better foundation for developing and analyzing models manually.

Modeling in AnyBody is done by a text-based input. For this purpose a special modeling language named AnyScript has been developed.

A text-based user input has been chosen for two reasons: (1) From the software developer's point-of-view it is easy to develop and maintain and (2) we believe that it is the only way to meet the goals of the AnyBody system mentioned in the Introduction. In particular, Goal 1 and 2 require a very versatile and flexible input.

The AnyScript Modeling Language

AnyScript is a declarative, object-oriented language for development of multibody dynamics models, particularly models of the musculoskeletal system. An AnyScript model is roughly divided into two main sections:

1) The model section containing the definition of the mechanical system, the body and the surroundings.
2) The study section containing lists of analyses and other operations to perform on the model. These can then be executed from the software.

The declarative nature of the language means that the language has a number of predefined classes that the user can create objects from. The predefined classes comprise (1) basic data types, such as numbers and string, (2) mechanical objects types such as segments, joints of various types, drivers, forces, and muscles, and (3) operational and model management classes.

The user cannot use operational code like 'do' loops and 'if-then-else' clauses and neither can new classes with additional functionality be defined[3]. Classes that function as containers for structuring the model hierarchically do however exist and they play an important role in structuring of large models.

The study section of the model allows for specification of various operations to perform on the model such as kinematical analysis, kinetic analysis, muscle calibration, and systematic parameter variations. Studies can refer to the entire model or to subsections of the model.

From a software design point-of-view, the definition of "studies" and "operations" as classes in the modeling language has enabled a clear relationship between the modeling data structure and the user interfaces of the software. Whenever as new "operation" or "study" is implemented in

[3] Future versions of AnyBody will get extended possibilities for user input of this kind.

the basic data structure, it immediately available both in the modeling language and from the user interface of the GUI.
Similarly, the graphical appearance of the model is programmed by the user in AnyScript by means of special objects known to the GUI. This puts the user in full control of view via the model input.

Structuring models for multiple purposes

Well-structured models are essential when creating large models such as full body models. AnyScript contains functionality for structuring models in a tree structure much similar to a file system. Folder-objects can be utilized to contain specific parts of the model and information can be accessed from many places through references. Typically, the data of large models is organized in several files and these can simply be combined by C-style include statements.
These features allow the user to structure the data according to the nature of the model and the practical use of the model. In particular, a model can easily be built by several modelers working on different files, and models and model parts can be exchanged.
It appears that Goal 1 and 2 from the Introduction, Page 2, have been reached. However, the system provides much freedom for the user for structuring the models in different ways, and this may prevent the interfacing of differently structured model parts with each other. If, for instance, one user has developed an arm model, and another user a hand model, it is likely that they will want to merge them, and different model structures might disable this option. To facilitate model merging, the AnyBody Research Group has developed a model structure, which splits the model into two distinct parts (Figure 2):

1) *The body model*. This part contains segments, joints, muscles, and other anatomical data, but no boundary conditions. Additionally it may contain setups for calibrating parameters of the body model, typically muscle parameters need adjustment for given body anthropometrics.
2) *The application model*. Application-specific data about movements, loads, and external parts, such as tools, bicycles, or the like, are placed in this part.

The idea is that users can exchange body models and connect them with different types of applications as illustrated in diagram of Figure 2.

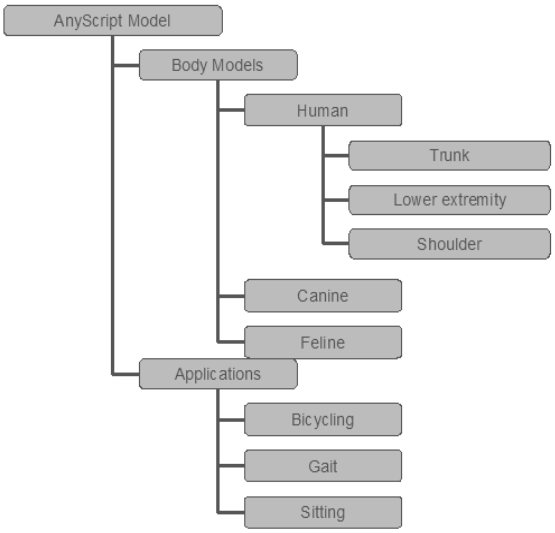


Figure 2: Recommended structure to facilitate model exchange. The labels in the boxes are only examples. Figure 3 shows a practical example.

As Figure 2 also illustrates, the body model itself is broken into parts. The idea of this is that different scientist can build models of different parts of the body and put them together as a virtual Frank-enstein's monster. This is important in view of the large effort that must go into building and validating good models.
The question then remains of how the elements that cross the interface between two body models or between a body model and the application can be handled. How can the developer of a hand be sure that the developer of the arm has provided muscle attachment points on the arm for the muscles spanning the wrist? To solve this problem the AnyScript language has been equipped with a facility to semantically allow addition of the necessary elements outside objects. This means that the hand model can contain the necessary additions to the arm model to make the parts compatible.

Conclusions

We have in this paper described the functionality and the principal considerations behind the AnyBody Modeling System; in particular, we have sought to explain how the four system design goals from the Introduction have been met. Meeting all four goals, we believe makes AnyBody a unique software system for the presented type of analysis. To our knowledge no other software offers similar inverse dynamic analysis capabilities together with general model building facilities.

Figure 3: A typical example of a structured model: The two bicycles are identical applications within the same model and are combined with a simple, 2-D lower extremity and a more complicated 3-D lower extremity respectively.

The efficiency of the muscle recruitment solver has made it possible to handle models of full body complexity. The current full body model, Figure 1, contains more than 400 muscles and a single time step can be solved in a few seconds on a standard PC. This allows for systematic parameter studies and optimization, which gives hope for computer-assisted design of man-driven machines, tools, and exercises like demonstrated in [19] and [20].

The scientific search for the "real" recruitment criterion is ongoing and maybe it will never be established. Neither of the presented criteria are based upon detailed knowledge about the real control system (the CNS), but rather upon an overall assumption about its optimal function. This may sound weak and less accurate than we are used to in typical engineering applications. The optimality approach does however fulfill basic conditions such as the dynamic equilibrium and the better of the criteria do provide physiologically reasonable results. Therefore, this should be considered as a reasonable approach while regarding the accuracy of the results with scrutiny.

Indeed biomechanical systems are generally difficult to handle accurately also in the experimental setups that are used to validate computational methods. Muscle activation and in particular muscle forces cannot be measured accurately and the nature of the system makes it impossible to measure all muscles. Naturally, this is a problem in the search for validation of computational models, but it also gives the models a special importance, since they in many cases are the only way to estimate certain valuable information. Many of the references for this paper contain comparisons of inverse dynamics methods to experimental results; for instance [21] and [22] contain results produced by AnyBody.

The optimality assumption and the use of inverse dynamics in general imply some restrictions on the use of the formalism. We neglect certain properties of the control system such as muscle activation dynamics and we assume optimality. We must therefore restrict the methodology to relatively slow and skilled motion, i.e., tasks that the human is familiar with. For unskilled tasks, one cannot expect the same degree of optimality. Whether the methodology can be extended to cover certain types of unskilled motion and more dynamical effects is indeed an interesting question.

Modeling humans and other creatures is a very demanding task and no single scientist or even institution could expect to accomplish this by themselves. Experiences from many years of research must be combined, and therefore it has been essential in the design of AnyBody to facilitate collaboration between scientists about model building. To meet this demand for model exchange, the AnyBody Research Group has made a public repository of models [23].

Acknowledgments

The development of AnyBody was supported by Aalborg University, grants from the Danish Research Council, and projects partially financed by Ford Werke and Ford FA, Germany. AnyBody Technology A/S [24] now distributes the system.

References

[1] Neptune RR. *Optimization algorithm performance in determining optimal controls in human movement analysis*. J. of Biomech. Engng. 1999; 121: 249-252.

[2] Anderson FC, Pandy MG. *A dynamic optimization solution for vertical jumping in three dimensions.* Computer Methods in Biomechanics and Biomedical Engineering 1999; 2: 201-231.

[3] Anderson FC, Pandy MG. *Dynamic optimization of walking.* J Biomech Eng. 2001; 123(5):381-90.

[4] Siemienski A. *Soft saturation, an idea for load sharing between muscles. Application to the study of human locomotion*. In Proceedings of the Symposium "Biolocomotion: a century of research using moving pictures", Cappozzo A, Marchetti M, and Tosi V (eds.), Promograph, Rome, Italy, pp. 293-303, 1992.

[5] Zajac FE. *Muscle and tendon: Properties, models, scaling and application to biomechanics and motor control*. Critical reviews in biomedical engineering 1989; 14: 359-411.

[6] Crowninshield RD. *Use of optimization techniques to predict muscle forces*. Journal of Biomechanical Engineering 1978; 100: 88-92.

[7] Pedotti A, Krishnan VV, Stark L. *Optimization of muscle force sequencing in locomotion*. Math. Biosciences 1978;38:57-76.

[8] Crowninshield RD, Brand RA*A. physiologically based criterion of muscle force prediction in locomotion.* Journal of Biomechanics 1981; 14: 793-801.

[9] Rohrle H, Scholten R, Sigolotto C, Sollbach W, Kellner H. *Joint forces in the human pelvis-leg skeleton during walking* Journal of Biomechanics 1984; 17: 409-424.

[10] Herzog W. *Individual muscle force estimations using non-linear optimal design.* J. Neurosci. Meth. 1987; 21: 167-179.

[11] Challis JH, Kerwin DG. *An analytical examination of muscle force estimations using optimization techniques.* Proc. Instn. Mech. Engrs. 1993; 207: 139-148.

[12] Dul J, Johnson GE, Shiavi R, Townsend MA. *Muscular Synergism–I. On criteria for load sharing between synergistic muscles.* Journal of Biomechanics 1984. 17: 663-673.

[13] van Bolhuis BM, Gielen CCAM. *A comparison of models explaining muscle activation patterns for isometric contractions.* Biological Cybernetics 1999; 81: 249-261.

[14] Rasmussen J, Damsgaard M, Voigt M. *Muscle recruitment by the min/max criterion – a comparative study.* Journal of Biomechanics, 2001; 34(3): 409-415.

[15] Damsgaard M, Rasmussen J, Christensen ST. *Inverse dynamics of musculo-skeletal systems using an efficient min/max muscle recruitment model*. In: Proceedings of IDETC: 18-th Biennial Conference on Mechanical Vibration and Noise, Pittsburgh, 2001.

[16] Forster E. *Predicting Muscle Forces in the Human Lower Limb during Locomotion.* Ph.D. thesis, Ulm University, Germany, VDI research report, 17, 241, Düsseldorf, VDI Verlag, 2003. (ISBN 3-18-324117-x)

[17] Bean JC, Chaffin DB., Schultz AB. *Biomechanical model calculation of muscle contraction forces: a double linear programming method.* Journal of Biomechanics 1988; 21: 59-66.

[18] Nikravesh PE. *Computer-Aided Analysis of Mechanical Systems*, Englewood Cliff, New Jersey: Prentice-Hall Inc., 1988.

[19] Rasmussen J, Damsgaard M. *Optimization of Biomechanical Multibody Systems*. In: Proceedings of The Second World Congress of Structural and Multidisciplinary Optimization, vol. 2, Gutkowski, W. and Mróz, Z., eds., Institute of Fundamental Technological Research, Warsaw, Poland, 1997 (ISBN 83-905454-7-0).

[20] Rasmussen J, Damsgaard M, Christensen ST, Surma E. *Design optimization with respect to ergonomic properties*. Structural and Multidisciplinary Optim. 2002; 24: 89-97.

[21] Thompson MS, Voigt M, de Zee M. *Evaluation of lower extremity musculo-skeletal model using sit-to-stand movement*. In: 19th congress of the International Society of Biomechanics, Dunedin, 6-11 July, 2003.

[22] Meijer, K., van Geenhuizen, J.A., Liebrand, H.T.J., Meijer, L.I., Pas, R.J., de Zee, M., Savelberg, H.H.C.M. *The functional consequences of muscle adaptation: a simulation in AnyBody.* In: Proceedings of the International congress on computational bioengineering, Zaragoza, Spain, 2003.

[23] The AnyBody Research Project and the AnyBody Repository, www.anybody.aau.dk.

[24] AnyBody Tech. A/S: www.anybodytech.com.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

CONTACT GRAPHS IN MULTIBODY DYNAMICS SIMULATION

Kenny Erleben* **and Henrik Dohlmann**
University of Copenhagen
Department of Computer Science
Denmark

Abstract

In rigid body simulation contact graphs are used detecting contact groups. Contact graphs provide an efficient underlying data structure for keeping information about the entire configuration and in this paper we extend their usage to a new collision detection phase termed "Spatial-Temporal Coherence Analysis". This paper will review contact graphs and demonstrate the performance impact in a typical constraint based multibody simulator.

Keywords: Multibody Dynamics, Contact Graphs, Contact Groups, Contact Analysis, High Level Control

Introduction

Historically contact graphs are used for splitting objects into disjoint groups that can be simulated independently. Contact graphs are frequently mentioned between people working with rigid body simulation, but they are often not formally described in the literature. In [1] the idea of using contact groups to break down contact force computations is mentioned. The benefit is obvious and not many people would spend time on explaining it. To our knowledge the first use of the word "graph" appears in [2], where a contact graph is used to properly back-up penetrating objects. In our opinion [3] is the first advanced attempt on using contact groups for distributed simulation. Recently [4] developed a shock propagation algorithm for efficient handling of stacked objects, which uses a contact graph. Today simulators do exploit contact groups for breaking down the computations into smaller independent problems as in Open Dynamics Engine (ODE) (v0.035), however they do not store a graph data structure.

Alternatives to contact graphs are not very surprisingly neither mentioned or talked about. An alternative is to put the contact-matrix into block-form [5]. In comparison with the contact graph approach the "block-form" matrix approach is limited to contact force and collision impulse computations and can not be used for anything else.

*Corresponding author. Phone: +45 3532 1413, Fax: +45 3532 1401, E-mail: kenny@diku.dk

The contact graph algorithm we present in this paper is part of the Spatial-Temporal Coherence (STC) analysis module. The algorithm clearly shows that STC analysis is scattered around the other phases in the collision detection engine. We use contact graphs for caching information, such as contact points. The cached information can be used for to improve run time performance of a rigid body simulator. Several speedup methods are presented, these fall into two categories, the first is real speed-ups due to improvements of simulation algorithms, the second is due to changes of the properties of the mechanical system, which alters the physical system, but still produces plausible results. Our main focus is computer animation and not accurate physical simulation.

The Contact Graph

A contact graph consists of a set of nodes, a node is an entity in the configuration, such as a rigid body or a fixed body. A node can also be a virtual entity, such as a trigger volume, generating an event notification when other objects penetrates it.

When objects interact with each other, contact information are computed and cached, it is easy to use the edges in the contact graph for storing information of interactions between objects. Edges are also useful for keeping structural and proximity information about objects.

The most important thing is that both nodes and edges should be accessible in constant time and edges are bidirectional and uniquely determined by the two nodes they run between.

These properties can be obtained by letting every entity in the configuration have an unique index, and letting edges refer to these indices, such that the smallest indexed entity is always known as *A* and the other as *B*.

An edge between a physical object and a trigger volume indicates that the physical object has moved inside the trigger volume. This kind of edges can therefore be used to generate trigger volume event notifications. This type of edge is a dynamic edge, meaning that it is inserted and removed dynamically by the collision detection engine during the simulation.

The last type of edges we can encounter are those which tell us something about how the objects in the configuration currently interact with each other. For instance if two rigid bodies come into contact then an edge is created between them. There are some combinations of edges which do not make sense. For instance an edge between two fixed bodies.

More node types and edge types are described in [6].

The Contact Graph Algorithm

We will now outline how a contact graph can be used in the collision detection pipeline. Notice that although we claim a contact graph to be a higher order contact analysis phase it is not a phase that is isolated to a single place in the pipeline, instead it is spread out around all the other phases, i.e. in between the broad phase module responsible for finding close object pairs, the narrow phase module that determines overlap status for an object pair, and the contact determination module that computes all contact points between an object pair. In the following we will walk through what happens in the collision detection pipeline step by step.

The first step in our algorithm is to update the edges in the contact graph. This is done by looking at the results of the broad phase collision detection algorithm. The results of the broad phase collision detection algorithm are really an unsorted list of pairs of nodes, where each pair denotes a detected overlap in the broad phase algorithm. Observe that each pair is equivalent to a contact graph edge. We can therefore insert new edges into the contact graph, which we have not seen before. At the same time we can handle all close proximity information, that is detection of vanished, persistent and new close proximity contacts. This is done by comparing the state of edges with their old state.

Now we can do logical testing and exploit caching, by scanning through all the reported overlaps and remove those overlaps we do not have or want to treat any further.

Overlaps with passive objects are also removed, passive objects do not really exist in the configuration, they are merely objects kept in memory in case they should be turned active later on. This way objects can be preallocated and further more there is no penalty in reallocating objects that dynamically enter and leave the configuration during runtime. We refer to the passive/active scheme as light weighted objects. The opposite is called heavy weighted objects and it means objects are explicitly deallocated and reallocated whenever they are added or removed from the configuration. One drawback of light weighted objects is that there is a penalty in the broad phase collision detection algorithm. Fortunately broad phase collision detection algorithms should have linear running time with very low constants, so the penalty is negligible.

The last screening test is for change in relative placement. Every edge stores a transform, $xform(\cdot)$, indicating the relative placement of the end node objects. If the transform is unchanged there is no need to run narrow phase collision detection nor contact determination, because these algorithms would return the exact same results as in the previous iteration.

We are now ready for doing narrow phase collision detection and contact determination on the remaining overlaps. Output from these sort of algorithms are typical a set of feature pairs forming principal contacts, *PCs* and a penetration state. The contact graph edges provide a good place for storing this kind of information. The output of the narrow phase should of course also be cached in the edge, because most narrow phase collision detection algorithms reuse their results from the previous iteration to obtain constant time algorithms. Notice that the closest principal contact is also determined, closest contacts are often used in impulse based simulation or time of impact estimation computations. We do also test for any contact state changes, that is if touching or pen-

etrating contacts vanishes or are persistent, that is if they were present in the last iteration. If one of the nodes were a trigger volume then we do not mark touching contact, but rather **in**- and **out**- events of the trigger volume, the same applies to the marking that took place earlier on.

Finally we can run the contact determination for all those edges where their end node objects are not separated.

In an impulse based simulator it is often not necessary to do a full contact determination only the closest points are actual needed [7], so an end user might want to turn of contact determination completely.

Also contact determination should be skipped on nodes representing things like trigger volumes, such entities are merely used for event notification, so there is no need for contact determination.

Now we have completed exploiting all of the logical and caching benefits we can gain from a contact graph. We are now ready for using the contact graph for its intended purpose, determining contact groups. The actual contact groups are found by a traditional connected components search algorithm, restricted to the union of the list of edges having survived the logical and coherence testing as described earlier and the structural edges. The algorithm works by first marking all edges that should be traversed as “white”. Afterwards edges are treated one by one until no more white edges exist.

Fixed objects are rather special, they behave like they have infinite mass, so they can support any number of bodies without ever getting affected themselves. They work like an insulator, which is why we ignore edges from these nodes when we search for contact groups.

The Event Handling

In the method we have described in this paper we have not really explicitly stated when events get propagated back to an end user. Instead we have very clearly shown when and how the events should be detected. We can traverse the edges of the graph, and simply generate the respective event notifications for all those edges that have been marked with an event.

There is one major subtlety to event handling, some simulators are based on backtracking algorithms, meaning they keep on running forward until things go wrong, then they backtrack correct things and tries to go forward again. This could many times. The consequences being that we might detect events which really never occurs.

The problem of backtracking can be handled in two ways, in the first solution events can be queued during the simulation together with a time stamp indicating the simulation time at which they were detected. Upon backtracking one simply dequeues all events with a time stamp greater than the time the simulator backtracks to.

In the second solution events are restricted to only be generated when it is “safe”, that means whenever a backtrack can not occur or on completion of the frame computation. We favor the second solution, the reason for this is that events are most likely to be used in a gaming context, where a backtracking algorithm is unlikely.

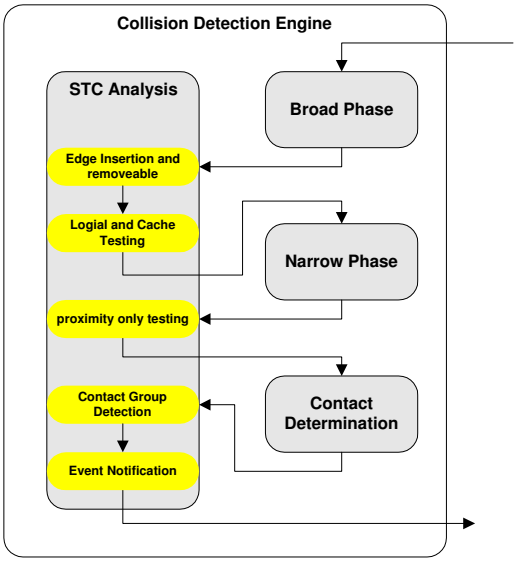


Figure 1: Spatial-Temporal Coherence Analysis Module.

The STC-Analysis Module

Having outlined how the contact graph should be used in the collision detection pipeline we can schematically sketch the STC analysis module together with the other modules in the collision detection engine. Figure 1 illustrates the interaction between the modules. From Figure 1 we see that the STC analysis occurs in three phases, post-broad-phase, post-narrow-phase and post-contact-

determination. We do not need a pre-broad-phase in the STC analysis at this point, but if any initialization is supposed to take place then a pre-broad-phase analysis would be a good place for doing this.

In our opinion there are basically three different ways to exploit the contact groups in rigid body simulation. We will briefly talk about them in the following.

Time warping: Traditionally one would backtrack the entire configuration when an illegal state is found, such as a penetration of two objects. This is inefficient since there may be a lot of objects whos motion are completely independent of the two penetrating objects. Contact groups could be used to only backtrack those contact groups with penetrating objects see [3] for more details.

Subdivision of Contact Force Computation: Constraint-based methods for computing contact forces are often *NP*-hard, so it is intractable to have large problems, however the contact forces needed in one contact group is totally independent of all the other contact groups. The essence is basically why solve one big problem, when you can solve several smaller problems instead see [1].

Caching Contact Forces: If contact forces from the previous iteration of contact force computations are cached in the contact graph edges then these forces can be used as initial guess for the contact force computation in the current iteration as described in [8].

Results

We will elaborate on several speedup methods that relies on or relates to contact graphs. The speedup methods are generally applicable to any kind of rigid body simulator. In order to show the effects we have chosen to extend our own multibody simulator, a velocity based complementarity formulation [9] using distance fields for collision detection, with the speed ups. Example code is available from the OpenTissue Project.

In this paper we will focus on performance speedup only. For this reason we have chosen a semi-implicit fixed time stepping scheme with a rather large timestep, 0.01 second.

Using distance fields for collision detection have one major drawback, when objects are deeply penetrating, a large number of contact points will be generated, the consequence will be a performance degradation due to the large number of variables. Performance improvements are therefore particular important even though real-time simulation is out of our grasp.

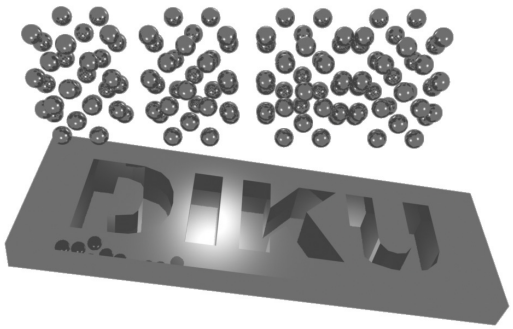


Figure 2: 120 Falling Spheres onto inclined plane with engravings.

We have done several performance measurements and statistics on 120 spheres falling onto an inclined plane with engravings. The configuration is shown in Figure 2. The total duration of the simulation is 10 seconds. In Figure 3 measurements of the brute force method is shown, i.e. without using contact graph. Observe that the number of variables and real-life time per iteration are increasing until the point where the spheres settle down to rest. In comparison Figure 4 shows how the curves from Figure 3 changes when a contact graph is used. Notice that the number of variables per contact group is much smaller than in the brute force method also observe the impact on the real-life duration curves.

The total running time of the brute force method is 28424 sec. Using a contact graph the simulation takes 1011.4 sec., which is a speedup factor of roughly 28. In the following we will explain 7 more speed up methods that further increases performance.

Using contact graphs an improvement comes from ignoring contact groups where all objects are at rest, we call such objects sleepy objects, and we determine them by tracking their kinetic energy, whenever we find an object who's kinetic energy have been zero within threshold over some constant number of iterations, which is user specified, the object is flagged as sleepy. If a contact group only con-

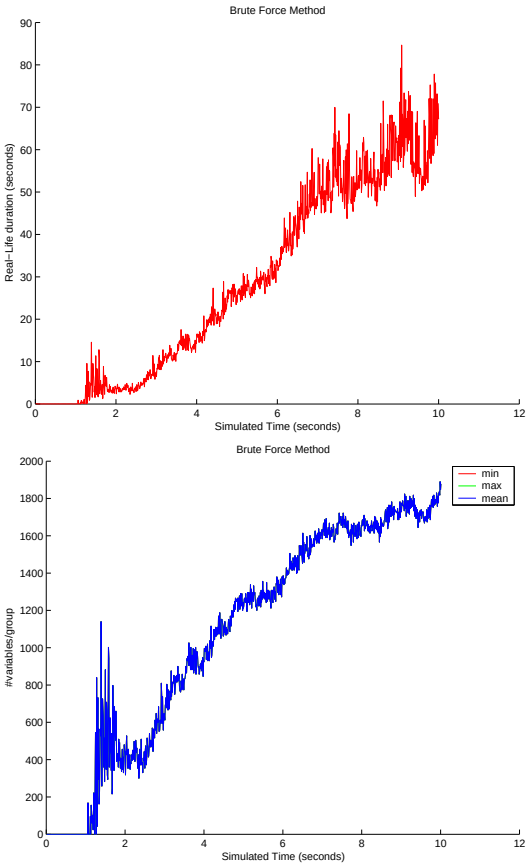


Figure 3: Performance of the brute force method.

tains sleepy objects the group is completely ignored during simulation. Contact graph nodes are used for the kinetic energy tracking. This method is computationally inexpensive, it have been used in all of the measurements in Table 1. The method could have a potentially disastrous effect if the scheme for tagging sleepy objects is not well-picked. Too greedy an approach could leave objects hanging in the air, too lazy an approach would result in no performance improvement.

We exploit contact graph edges for caching contact points and contact forces. Cached contact points are used to skip narrow phase and contact determination whenever two incident objects of a contact edge are at absolute rest, cached contact forces are used to seed the iterative LCP solver, Path from CPNET. We call this speed up “caching”.

A further speed up can be obtained by limiting the number of iteration of the LCP solver, currently from 500 to 15, as a consequence the motion is altered but still looks plausible. The method has noth-

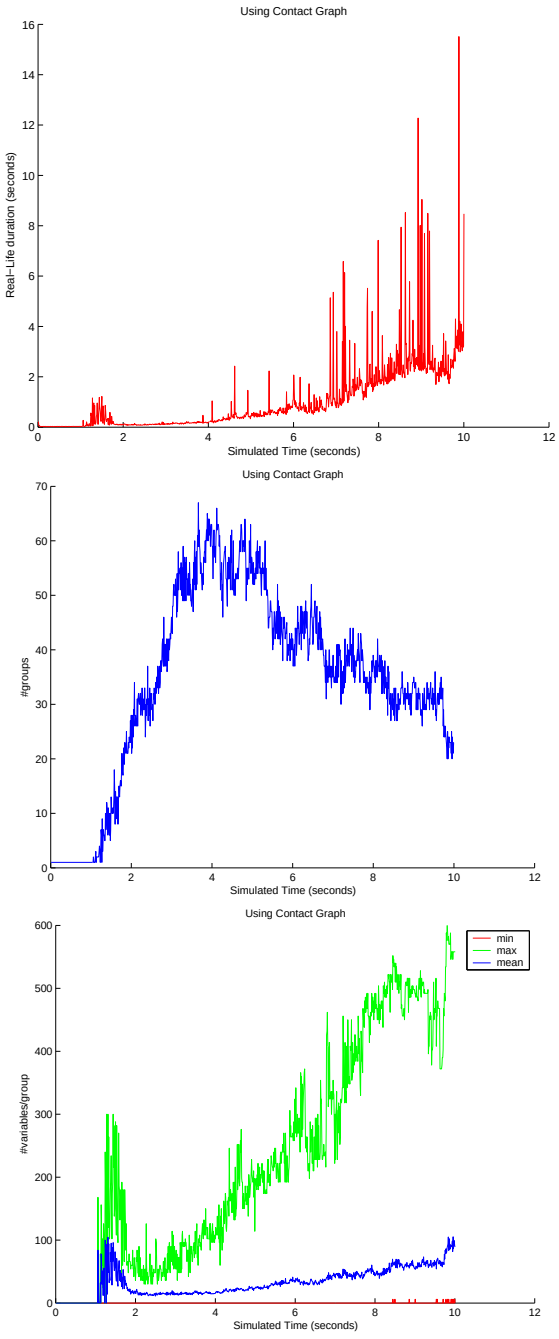


Figure 4: The performance impact of using a contact graph.

ing to do with contact graphs, but it is interesting in combination with the other speed ups we apply. We refer to this speed up as “tweaking”.

Another speed-up we use is to reduce the number of contacts between two objects in contact, the reduction is applied to objects that are deeply penetrating, all contacts are pruned except the single contact of deepest penetration. The contact graph edges are a convenient storage for this. We have named this “reduction”. Reduction have an effect on the motion of the objects, we believe that it is actually more correct, because intuitively the deepest point of penetration better resembles the original idea behind using the minimum translational distance as a separation measure. Besides theoretically reduction should decrease the number of variables used in the complementarity formulation.

Inspired by the speed up of detecting independent contact groups, further subdivision into independent groups seems attractive. An idea is to prune away sleepy objects from those contact groups containing both non-sleepy and sleepy objects, thereby hopefully breaking these into subgroups. We refer to this as “subgrouping”. Other subgrouping/sleepy object schemes can be found in [5, 10]. To help objects settle down and become sleepy faster it intuitively seems to be a good idea to let the coefficient of restitution fall to zero the more sleepy an object gets, meaning that sleepy objects are sticky objects. Currently we set the coefficient of restitution to zero whenever at least one of the incident objects are sleepy. We call this “zeroing”. In the same spirit a linear viscous damping term is added to the motion of all objects in the simulation, the intention is to slow down objects making them less willing to become non-sleepy. We call this heuristic “damping”. The contact graph is used for the subgrouping and zeroing by classifying edges dependent on the sleepy-state of the incident objects.

The last method we have applied consist of setting the inverse mass and inertia tensor to zero for all sleepy objects. The main intuition behind this is to “force” sleepy objects to stay sleepy. We have named this “fixation”. It has a dramatic impact on the simulation as seen form the ♠-simulation in Figure 5. Fixation makes only sense when subgrouping is used otherwise the LCP solver will have to solve contacts between two fixated objects.

Table 1 contains performance measurements of most promising combinations of the previously described heuristics and speedup methods.

	Cache	Tweak	Reduce	Zero	Damp	Subgroup	Fixate	Time
♢	+	-	+	-	-	-	-	624.274
♡	-	+	-	-	+	-	-	528.633
♣	+	-	-	-	+	+	-	460.561
♠	+	+	+	-	+	+	+	134.721

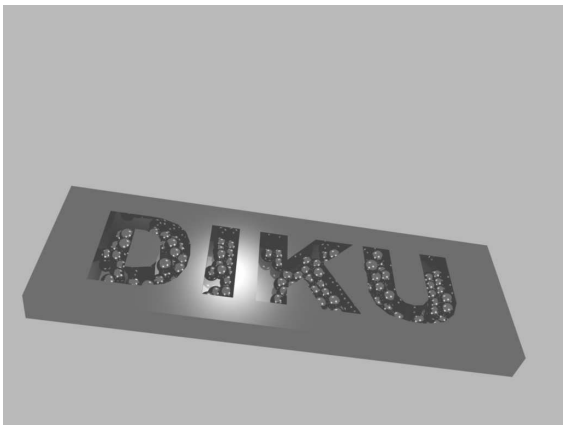
Table 1: Comparison of various combinations of speed-up methods. “+” means enabled, “-” means disabled. More combinations can be found in[6].

Figure 5 shows motion results of the two combinations: ♢ and ♠. Here ♢ is identical to the brute-force method. These four combinations: ♢, ♡, ♣, and ♠ were picked because they resembles the best performance. It should be noted the motion diverges more and more from the brute force method the more speed ups that are used. Especially ♠ is different, during the last seconds objects actually fly up in the air. Animations of the ♢, ♡, ♣, and ♠ simulations are available from corresponding authors homage. In Figure 6 a comparison is done between the performance statistics of ♣, and ♠. The ♢ and ♡ behave similar to ♣ as can be seen in [6]. The plots of ♣ are similar to Figure 4. The ♠, has very different plots for the real-life duration and variables per group plots, these appear to be nearly asymptotically constant.

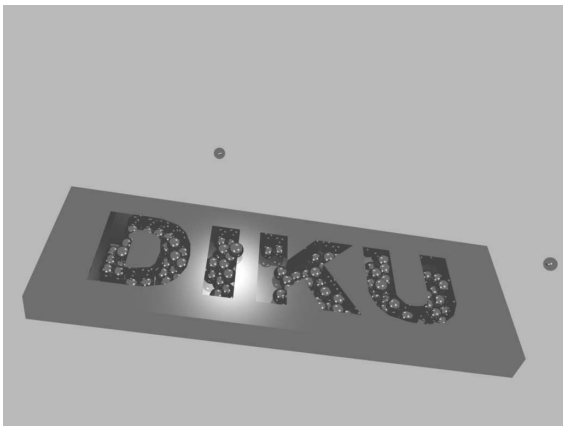
Discussion

It is obvious from Table 1 that the prober combination of the speed ups is capable of producing a speed up factor of $\frac{28424}{135} \approx 210$. It is difficult to describe the impact on the resulting motion, however it is clear that using Contact Graphs, Caching Contact Forces and Sleepy Groups do not alter the mechanical system, but the other speed ups presented change the physical properties and thus the motion of the objects as can be seen in Figure 5. Especially the reduction and the subgrouping also with zeroing and fixation have great impact on the motion.

The tagging of sleepy objects can have a drastic impact on the simulation. As an side effect objects are sometimes left hanging. For instance in Figure 7 spheres land on top of each other, while the top-most spheres rumbles off, the bottom-most sphere is kept in place and prohibited from gaining kinetic energy, at the end of the simulation a single hanging sleepy sphere can be seen near the K-letter. We have to be



(a) ♢



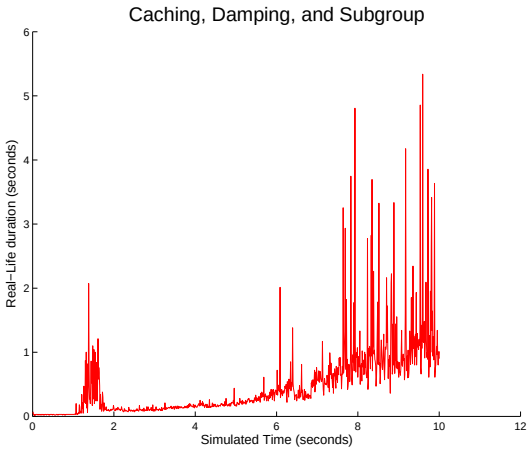
(b) ♠

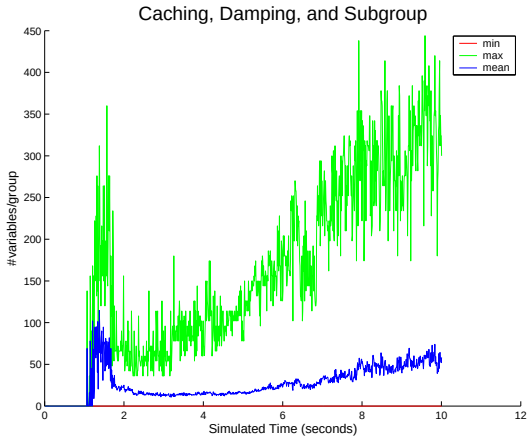
Figure 5: Motion Results at time 9.20 seconds for ♢ and ♠.

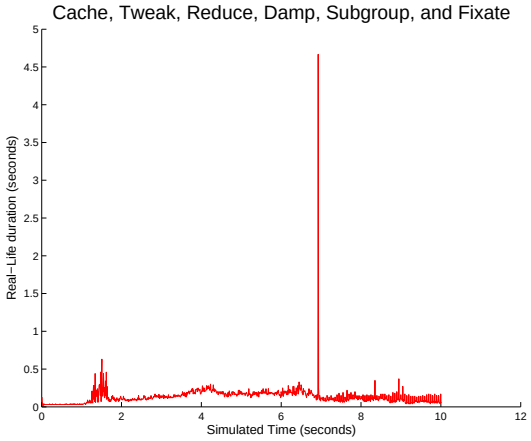
careful not making general conclusions based on the measurements in this paper, since only one configuration have been examined.

We can say as much as contact graphs are a valuable extension to a rigid body simulator, even when not trading accuracy for performance speed up factor of order 20-30 is not unlikely, disregarding accuracy the speed up factor can be increased further by an order of magnitude.

Better performance is not always achieved by using more speed ups, in some cases one speed up cancels the effect of another. For instance using caching seems to make tweaking needles. Also speed ups can alter the motion. Thirdly our experiments indicate that high-performance can be achieved by combining subgrouping and fixation, however from the motion results it also clear this is a non-trivial task to embark upon.







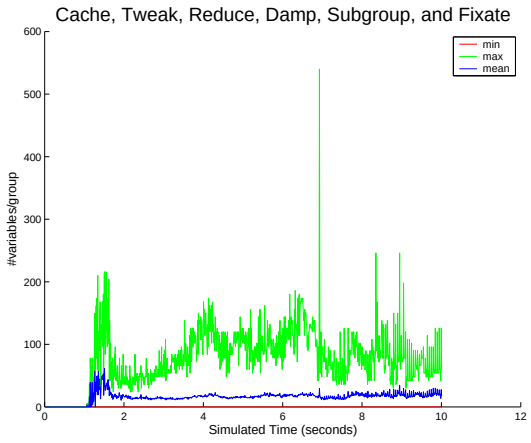


Figure 6: Performance measurements of ♣, and ♠ from Table 1.

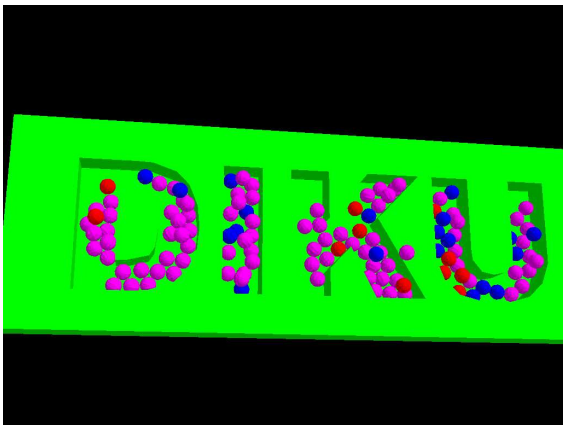


Figure 7: Figure showing hanging sphere near K-letter. Frame grab after 9.87 sec. using zeroing, damping, subgrouping and fixation.

Our numerical experiments clearly indicates that sleepy objects are a promising strategy, it seems promising to look into better methods for more quickly making objects sleepy and stay sleepy. For instance to pre-process the complementarity formulation with a sequential collision method truncating impulses, this was used for a sequential impulse based method in [4].

References

[1] Anitescu M, Potra F.A, Stewart D. *Time-stepping for three-dimensional rigid body dynamics*, Comp. Methods Appl. Mech. Engineering, 1998.

[2] Hahn J. K. *Realistic animation of rigid bodies*, In Computer Graphics volume 22, 1988.

[3] Mirtich B. *Timewarp rigid body simulation*, In Proceedings of the 27th annual conference on Computer graphics and interactive techniques, 2000.

[4] Guendelman E, Bridson R, Fedkiw R. *Nonconvex rigid bodies with stacking*, ACM Transaction on Graphics, Proceedings of ACM SIGGRAPH, 2003.

[5] Barzel R. *Physically-based Modelling for Computer Graphics, a structured approach*, Academic Press, 1992.

[6] Erleben K. *Contact graphs in multibody dynamics simulation*, report no. 04/06, Copenhagen, DIKU, 2004.

[7] Mirtich B. *Impulse-based Dynamic Simulation of Rigid Body Systems*, PhD thesis, Berkeley, University of California, 1996.

[8] Baraff D. *Dynamic simulation of non-penetrating Rigid Bodies*, PhD thesis, Cornell University, 1992.

[9] Stewart D, Trinkle J.C. *An implicit time-stepping scheme for rigid body dynamics with inelastic collisions and coulomb friction*, International Journal of Numerical Methods in Engineering, 1996.

[10] Schmidl H. *Optimization-based animation*, PhD thesis, University of Miami, 2002.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

SELF INTERSECTION WITH CULLIDE

Jonas Meyer *
Niels Boldt†
Department of Computer Science, University of Copenhagen, Denmark

Abstract

We present an image-space technique which can detect intersections and self-intersections among multiple moving and deforming objects. No preprocessing is needed and the shape of the objects are unconstrained and can be an arbitrarily polygonsoup. Compared to other intersection detection algorithms running on graphics hardware the algorithm only make modest use of bandwith between the CPU and GPU because no buffer readbacks are necessary.
Keywords: Collision Detection, Image-Space, Graphics Hardware, Self Intersections

Introduction

Efficient collision detection is a fundamental problem in physical-based simulation, computer animation, surgery simulators and virtual prototyping and can often be the bottleneck in the mentioned areas. For rigid bodies many efficient approaches, based on pre-computed bounding-volume hierarchies have been proposed [Hub93, GLM96].

But as deformable objects are becoming normal and necessary, these hierarchical data structures can not be precomputed, but must be updated frequently. Some effort has been given to optimize this update [LAM01], but it is still expensive to update the hierarchical data structures in dynamic environments. An even harder problem is posed if self-intersections in a deforming object must be detected. Use of bounding-volume hierarchies must be abandoned, because traversal of the hierarchies will be slow due to the many overlaps of bounding volumes. For cloth simulation acceptable solutions has been proposed [VMT00, VT95].

In this paper, an image-space technique for intersection detection of arbitrarily shaped, deforming object is presented. The objects can deform in any way and both open and closed objects can be treated. The approach is based on Cullide, presented in [GRLM03], but in contrast, we address, and solve the problem of self-intersections.

Compared to previous approaches, our approach is quite simple and can easily be implemented on commodity graphics hardware. As processing power of graphics hardware grows faster than processing power of general purpose hardware, our approach should be able to scale faster than algorithms implemented on standard CPU's. There is no restriction on the geometry used, except that it should be triangulated. Neither preprocessing nor hierarchical structure are necessary for the correctness of the algorithm. However, a hierarchy based on mesh connectivity can improve the algorithm significantly.

The rest of the paper is organized as follows: We give a description of previous work done on algorithms for detecting self-intersections and intersections using graphics hardware in Previous Work. We give an overview of the original Cullide algorithm in Cullide. We present our modification to Cullide in Self Intersection. In Precision Issues we discuss problems with image-space precision and compare precision of Cullide and the modified version. In Results, various tests are performed, to rate the performance of the modified algorithm. We finish the paper with a conclusion and directions for future work.

Previous Work

In this section we will give a brief review of prior work on image-space techniques for intersection detection.

An approach to image-space intersection detection is presented in [BWS99]. The depth layers of two convex objects are rendered into the depth buffer.

*email: meyer@diku.dk
†email: boldt@diku.dk

Any overlaps between the objects is found using the stencil buffer. The method can only handle convex objects and do not consider self-intersections. The main problem is that the stencil buffer must be read back into main memory and processed.

In [HKL+99], graphics hardware is used to compute generalized 2D and 3D Voronoi diagrams, which can be used for proximity queries. For each sample point, closest site and distance to that site is computed, using polygon scan-conversion and depth buffer comparisons. Distance meshes for points, lines, curves and curved surfaces in 2D and 3D can be computed.

A specialized approach to intersection detection for cloth on walking people is presented in [VSC01]. Depth map, normals and velocity for the avatar is computed in image-space. Intersection detection for a particle can be done by transforming to image space and comparing with the depth map. The algorithm relies heavily on buffer read back.

In [HTG04] an image-space technique is used to compute a layered depth image, which can be queried for collisions and self-intersections. The algorithm requires that the geometry is a twofold. The algorithm is implemented in two versions, on CPU and in graphics hardware. For small configurations the CPU-implementation is most efficient. For big configurations with up to 500k faces the GPU-implementation running on graphics hardware is most efficient. Again, the algorithm relies on buffer read back.

In [GRLM03], occlusion queries are used to prune a set of potentially overlapping triangles. No preprocessing is needed and arbitrary polygon soups can be handled. An advantage is that buffer read backs are not needed, implying minimal use of bandwidth.

Cullide

In this section we will give an overview of Cullide as presented in [GRLM03]. An object is defined to be a collection of one or more triangles. In the following we assume an environment O of n objects $o_1, o_2, \ldots, o_n$.

In the following, we assume that all objects are divided into some form of hierarchy. We will ignore the form of the hierarchy because it only influences performance and not correctness.

Overview

Given a collection of objects, the purpose of a collision detection algorithm, is commonly to determine either of the following

- A set of pairs of intersecting triangles.
- A set of pairs of closest features.

Most collision detection algorithms determine either one or both of the above.
Cullide is different in this aspect, as it does not directly determine any of the two results mentioned above. Instead, the algorithm takes as input a set O, of n objects. The algorithm reduces this set, and returns a new set, called the PCS, *potentially colliding set*, which is a subset of the original set.
For clarity, we consider Cullide consisting of two parts. The first part manages the graphics hardware and controls what should be sent to it. It also controls the usage of object hierarchies. The second part is a simple operation, `reducePCS`, which can be implemented using graphics hardware. The operation take as input a set of objects O, and prune some of the non-intersecting objects away. The operation does not guarantee that all non-intersecting objects will be removed, but it does guarantee that no intersecting objects will be pruned from the set.
The `reducePCS` operation can be considered a primitive operation - There are few variations on to how it can be implemented. The performance of the algorithm depends primarily on the first part of the algorithm - we can not improve `reducePCS`, because it depends solely on the performance of the graphics hardware.

Part 1: Usage of the object hierarchies

The usage of the hierarchies can be relatively simple. We start by running `reducePCS` at the object level, pruning non-intersecting objects. We then replace all the non-leaf objects in the PCS, with their children and repeat the operation. The algorithm is finished once the PCS is empty or all the objects in the PCS are leaf nodes. The following pseudo code describes the algorithm:

Part 2: The `reducePCS` operation

The `reducePCS` operation is implemented on graphics hardware using occlusion queries, see

```
cullide( Objects )
  PCS = Objects
  do
    PCS = reducePCS( PCS )
    for each node in PCS
      if !is_leaf( node )
         PCS.remove( node )
         PCS.insert( node.children )
  while has_non_leaf_nodes( PCS )
  return PCS
```

Figure 1: A basic version of Cullide.

[GRLM03] for elaboration of the details.
The visibility information gained from occlusion queries is used to prune objects from the PCS. If all fragments pass the depth test, when rendering an object, the object is said to be fully visible, with respect to the set of objects already rendered to the depth buffer. Thus we get the first lemma:

Lemma 1. *If all fragments generated, when rendering an object, pass the depth test, the object is fully visible. When an object is fully visible, the object does not intersect with any of the objects already rendered to the depth buffer.*

The proof is trivial. If there is an intersection with any of the objects already rendered, the intersection will cause some of the pixels to fail the depth test.
An object can be pruned from the PCS if it is determined that the object is fully visible, with respect to all other objects in the set. A naive approach to using the algorithm would require rendering each object n times, thus yielding a time complexity proportional to $\mathbf{O}(n^2)$. The solution to the problem lies in the following lemma

Lemma 2. *Given a set, O, of objects $o_1, o_2 \dots o_n$. The object o_k , $k \in [1,n]$ is fully visible with respect to $O \setminus o_k$, iff it is fully visible with respect to $O'_k = o_1 \dots o_{k-1}$ and fully visible with respect to $O''_k = o_{k+1} \dots o_n$.*

Again, the proof is trivial. If o_k is fully visible to both O'_k and O''_k, it must be fully visible to the union of the two sets, $O \setminus o_k$.
Using lemma 2, we see that the two subproblems exhibits an optimal substructure. That is, when testing object o_k, against the set O'_k, we note that $O'_k = O'_{k-1} \cup \{o_{k-1}\}$. Thus we can reuse the rendered data. Likewise for O''_k.

To take advantage of this, we first test all objects against their respective O'_k, and then test all objects against their O''_k. The effect of this, is that the running time of the algorithm is reduced to $\mathbf{O}(n)$. Pseudo code for the `reducePCS` algorithm is given in figure 2.

```
reducePCS( Objects )

  ClearDepthBuffer()
  for each object in Objects
    DepthTest( GREATER_EQUAL )
    DepthMask( FALSE )
    BeginOcclusionQuery()
    object.render()
    object.fullyvisible =
      EndOcclusionQuery() == 0
    DepthTest( LESS )
    DepthMask( TRUE )
    object.render()

  reverse( Objects )

  ClearDepthBuffer()
  for each object in Objects
    DepthTest( GREATER_EQUAL )
    DepthMask( FALSE )
    BeginOcclusionQuery()
    object.render()
    object.fullyvisible =
      EndOcclusionQuery() == 0
      && object.fullyvisible
    DepthTest( LESS )
    DepthMask( TRUE )
    object.render()
  remove_fully_visible( Objects )
  return Objects
```

Figure 2: Pseudo code for reducePCS. As `EndOcclusionQuery()` returns the number of fragments passing the inverted depth test, a object is fully visible with respect to the objects rendered into the depth buffer if this query returns 0.

Self-intersections

A drawback of Cullide, is that the algorithm is unable to detect self-intersections. This is due to the construction of the `reducePCS` operation where object o_k only are tested for intersection against the other $n-1$ objects. So when we remove an object from the PCS, we know that it does not intersect

with other objects, but we do not know if any self-intersection exists in the removed object. In the following we will explain how to modify Cullide such that the algorithm becomes capable of detecting self-intersections.

Algorithm

A simple idea, would be to test the object against itself, while testing against either O'_k or O''_k. This is not immediately possible, since the depth test function is less than. If we render an object to the depth buffer, and then render it again, doing an occlusion query, it will fail no matter what, since none of the fragments will generate lesser depth values. This happens because we submit the same object twice - during the second pass, the rasterizer will generate exactly the same fragments.
This can easily be remedied, by changing the function to less than or equal. This ensures that *some* of the fragments will pass the depth test. We thus have an opportunity to prune away subparts of the object. Faces closest to the viewer will be prune able. Faces obscured, fully or partially, by other faces can not be pruned before the faces in front of them are pruned.
In order to see how we detect self-intersections, we must consider the objects visibility as a whole. We describe this visibility with reference to the object itself as self-visibility. The following definition formalizes it:

Definition 3. *An object is fully self-visible, if all fragments generated by rendering the object to a depth buffer in which the object has already been rendered, pass the modified depth test.*

The crucial observation, is similar to what appeared in the original Cullide algorithm - Fully visibility means no intersections. We restate it regarding self-intersections and self-visibility:

Lemma 4. *An object which is fully self-visible never has any self-intersections.*

The proof of this lemma is obvious. If an object is fully self-visible all fragments generated can be seen if we look in the direction of projection. A necessary condition for self-intersection is that a fragment generated must be hidden behind another fragment. Therefore we conclude that no self-intersection can exist. Notice that the lemma above is vague - fully

```
reducePCS( Objects )

  ClearDepthBuffer()
  for each object in Objects
    DepthMask( TRUE )
    DepthTest( LESS )
    object.render()
    DepthMask( FALSE )
    DepthTest( GREATER )
    BeginOcclusionQuery()
    object.render()
    object.visible =
      EndOcclusionQuery() == 0

  reverse( Objects )

  ClearDepthBuffer()
  for each object in Objects
    DepthMask( FALSE )
    DepthTest( GREATER )
    BeginOcclusionQuery()
    object.render()
    object.visible =
      EndOcclusionQuery() == 0
      && object.visible
    DepthMask( TRUE )
    DepthTest( LESS )
    object.render()

  remove_fully_visible( Objects )
  return Objects
```

Figure 3: Our modified version of `reducePCS` where objects containing self-intersection are not pruned from the PCS.

self-visibility implies no self-intersections, but objects that are not fully self-visible are not necessarily self-intersecting.
We need to expand our rule to sub-objects, otherwise, we will be unable to prune many objects. The following lemma describes self-visibility recursively, as a function of the self-visibility of the sub-objects. This allows recursive testing, and pruning objects by the use of their hierarchies.

Lemma 5. *An object is fully self-visible, if all its sub-objects are fully self-visible and fully visible with reference to all other sub-objects.*

Again, the proof is trivial. If an sub-object does not intersect with any other sub-object, and neither intersects with itself, it can not be part of any self-intersection. If this applies for all sub-objects, the

object can not contain any self-intersections.
Turning back to the original version of `reducePCS` (Fig. 1), we can see that it does already test all sub-objects for visibility against each other - We must therefore modify it to test against itself.
By changing the depth function and adding a test against the object itself, while testing against O'_k, we get a modified version of `reducePCS`, which does not prune self-intersecting objects. This is the only change necessary, to enable Cullide to handle self-intersections. The modified version is shown in figure 3.

Drawbacks

There are drawbacks to the method. The first comes from the change of depth function. The original Cullide algorithm was able to detect contacts. Our method can not do this, but is limited to detecting penetrations. This limits our method to detecting penetrations. We do not believe that it will be a problem. The primary reason is, that there are no way of distinguishing between contacts and penetrations. If an algorithm is to use our method (or the original Cullide), it would have to utilize some algorithm to distinguish between faces penetrating, and faces in contact. Further, when detecting contacts, algorithms provide some means to set a minimum detection distance, a collision envelope, such that all primitives closer than this minimum distance are treated as contacts. This is not possible with Cullide either, but it is implicitly set, by the precision of the depth buffer and the resolution of the screen.
Another drawback appears when considering closed objects. It should be obvious, that all closed objects will *not* be fully visible with respect to themselves - Some object parts are bound to obscure other parts. Thus they are guaranteed to stay in the PCS. This does not affect the correctness of the algorithm, since the algorithm descends to the lower levels of the hierarchy of the object, and prunes on the lower levels of the objects.

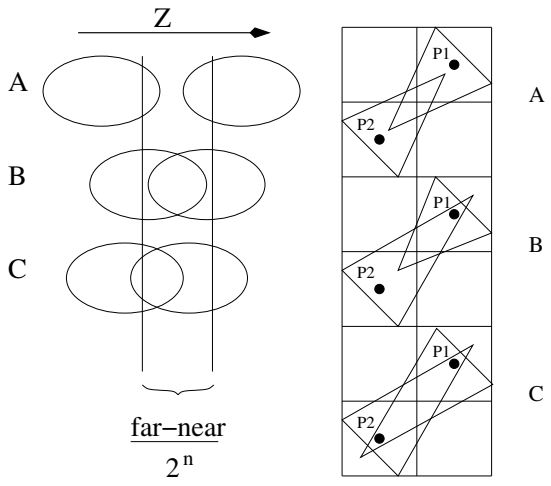
Precision Issues

In order to correctly utilize the algorithm, it is necessary to understand the implications of using the graphics hardware. In this section we will shortly explain what has an effect on the precision of the algorithm. The discussion will generally apply to the original Cullide and our version, and we will mention when they differ.
The algorithm can be considered as a hardware accelerated discrete approach to a brute force solution. Thus the precision issues with the algorithm occurs when transforming our exactly formulated problem into a discrete one. This happens when using the rasterizer of the graphics hardware.
The first problem is regarding the rounding of the values used in the depth buffer. Here our approach differs from Cullide. In Cullide, the strict less than test ensures that if two polygons map to the same depth, they will generate an intersection. Since our method changes the depth test function, it requires objects to penetrate by a certain amount, before the depth test will ensure detection of intersections. Given an orthographic projection with clipping planes given by $near, far, far > near$, and a depth buffer with n bits of precision, they need to penetrate by at least

$$\frac{far - near}{2^n}$$

to guarantee detection.

In figure 4(a), three different cases of overlap in depth values are illustrated. The vertical lines indicate where the rounded value changes. There is no overlap in A, and penetration on both B and C.



(a) depth value rounding (b) rasterization

Figure 4: Difference in precision between Cullide and our method

Figure 5: This figure shows our deformed Stanford bunny. The intersecting triangles are shown with green.

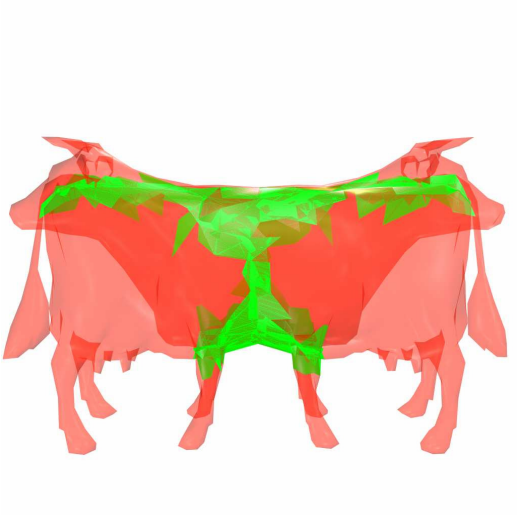


Figure 6: This figure shows our intersecting cows. The intersecting triangles are shown with green.

Cullide detects an intersection in all three cases. Our algorithm only detects an intersection in case C.

The second problems is regarding to the rasterizer. The general rule that rasterizers follows, is to fill only pixels having centers completely inside the triangle being rendered. This has the effect, that triangles can overlap in screen-space, without actually filling the same pixels. The bound for this overlap is, given a screen with a pixel of size $width \times height$, $\sqrt{width^2 + height^2}$.

Figure 4(b), shows 12 pixels of the screen that we render three cases to, here illustrating the problems with the resolution of the screen. The dots mark the center of the pixels. In all three cases, the triangles intersect. In case A, neither Cullide or our method will detect an intersection. In case B, Cullide will detect an intersection iff the triangles depth values at P1 are rounded to the same value. Our method will not detect an intersection in case B. In case C Cullide will always detect an intersection, our method will detect one if the distance between the depth of the fragments at P1 and P2 are at least $\frac{far-near}{2^n}$.

Given a triangle, we observe, that the closer the normal of the triangle is to being perpendicular to the direction of projection, the smaller will the projected triangle be. In fact, a triangle with normal perpendicular to the direction of the projection should not generate any pixels at all. This does unfortunately cause erroneous pruning of triangles.

Results

We have implemented our system on a 2.8 GHz Pentium 4 with a GeForceFX 5900 Ultra graphics card. We have tested our implementation, using modified meshes from [Bra], in the following environments:

- A deformed Stanford Bunny with a single intersection area. We have tested with triangles count from 5000 triangles to 270000 triangles. We are able to prune these models in 20 to 600 ms. The model is shown in figure 5.
- Two cows combined to one object such that many self-intersections exists. We have tested with triangle count from 3000 to 270000 triangles. We are able to prune this model in approximately 35 ms to 1000 ms. The model is shown in figure 6.

A comparison of pruning times is presented in figure 7. This is done at a resolution of 500x500. It can be seen that it is significantly faster to prune the bunny compared to the cow. This is caused by the large amount of self-intersections in the cow, which make the algorithm descend deeply into the hierarchy.

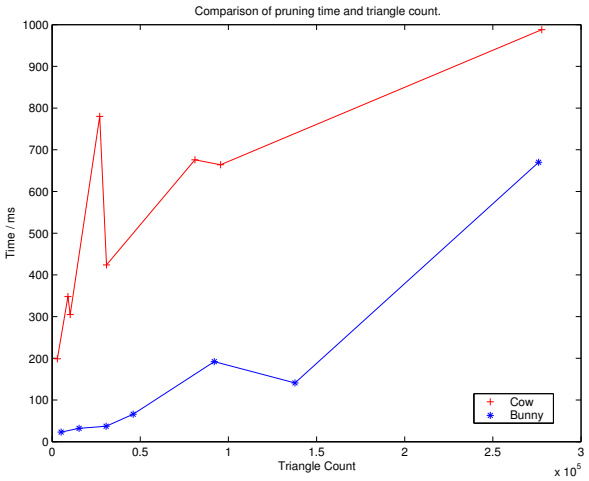


Figure 7: A comparison of pruning time for bunny and cow with varying triangle count. The peak for the cow is due to an extra number of passes required to prune the PCS compared to other instances of this model.

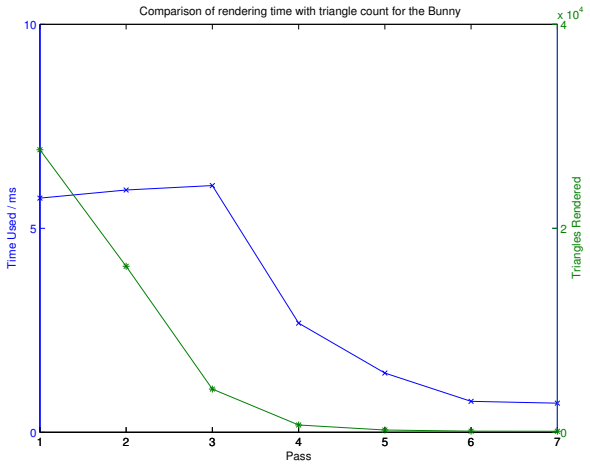


Figure 8: The time used and triangles rendered is plotted, for each pass, for the bunny with 5000 triangles. The used time and number of triangles rendered does not necessarily depend linearly on each other, because the number of occlusion queries differ in the passes.

In figure 8 we have compared pruning time with triangles rendered for each pass of a single collision query on the cow and bunny. A pass includes several calls to `reducePCS` with different direction of projections. In figure 9 we compare the relative performances of the two collision queries on the cow respectively the bunny. For the cow approximately 40% of the used time, is spent pruning 2-3% of the pruned triangles. The same tendency is exhibited with the bunny.

Performance Analysis

From the test results we can conclude that the performance of the algorithm depends on triangle count, number of occlusion queries, and number of intersections. Another important observation, we made during development of the modified Cullide is the importance of the hierarchy. Initially we used a randomly built hierarchy, which performed poorly. Instead we used a hierarchy built on mesh connectivity, which improved performance significantly. All of the tests are based on hierarchies built this way.

Conclusion and further work

We have presented Cullide, and shown how to augment it, to handle self-intersections.

The modified Cullide seems reasonably easy to implement, but this is not the case if performance is of importance. To maximize performance of the modified Cullide clever strategies for rendering and construction of hierarchies must be made.

Performance wise many things could be done. Systematic methods for finding the best direction of projection could be based on assumptions of temporal coherence or object orientation. A heuristic for finding the time, at which pruning should be turned to the CPU can improve the algorithm.

Decomposition of objects affects the performance of the presented algorithm significantly and therefore different strategies for decomposition of objects could be examined. This could also be examined in the context of deformable objects where deformations can affect the decomposition.

Regarding the precision issues, some work is necessary to make the algorithm stable, and to ensure that no intersecting triangles can be pruned. Testing how close a triangles normal is to being perpendicular to the direction of projection, could be used to estimating which triangles that are too small to be pruned. This is definitely possible, but we believe it will be hard to implement on current graphics hardware, without imposing some

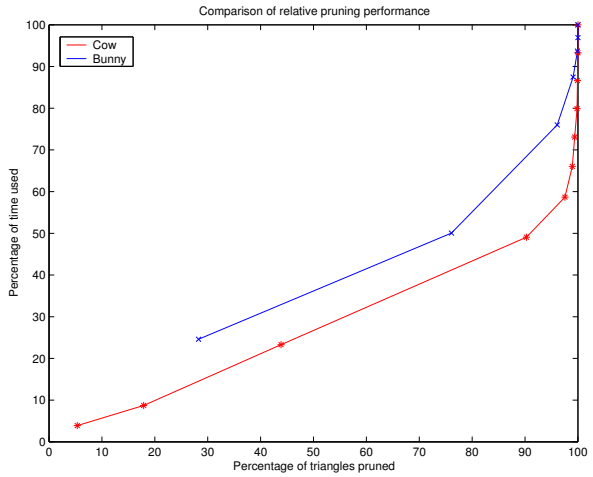


Figure 9: A comparison of the relative pruning performance for a occlusion query on the cow and bunny. On the x-axis the accumulated percentage of triangles pruned can be read, while the percentage of the accumulated used time can be read from the y-axis. The starting point of the lines denote the percentage of triangles pruned in first pass.

restrictions to the layout of the triangles. Future graphics hardware is very likely to provide capabilities, allowing the process to be fully automated.

References

[Bra] Gareth Bradshaw. http://isg.cs.tcd.ie/spheretree/.

[BWS99] George Baciu, Wingo Sai-Keung Wong, and Hanqiu Sun. Recode: an image-based collision detection algorithm. *The Journal of Visualization and Computer Animation*, 10:181 – 192, december 1999.

[GLM96] S. Gottschalk, M. C. Lin, and D. Manocha. OBBTree: A hierarchical structure for rapid interference detection. *Computer Graphics*, 30(Annual Conference Series):171–180, 1996.

[GRLM03] Naga K. Govindaraju, Stephane Redon, Ming C. Lin, and Dinesh Manocha. Cullide: interactive collision detection between complex models in large environments using graphics hardware. In *Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware*, pages 25–32. Eurographics Association, 2003.

[HKL+99] Kenneth E. Hoff III, John Keyser, Ming Lin, Dinesh Manocha, and Tim Culver. Fast computation of generalized Voronoi diagrams using graphics hardware. *Computer Graphics*, 33(Annual Conference Series):277–286, 1999.

[HTG04] Bruno Heidelberger, Matthias Teschner, and Markus Gross. Detection of collisions and self-collisions using image-space techniques. *WSCG*, 12(1-3), februrary 2004.

[Hub93] Philip. M. Hubbard. Interactive collision detection. In *IEEE Symposium on Research Frontiers in Virtual Reality*, pages 24–32, 1993.

[LAM01] Thomas Larsson and Tomas Akenine-Möller. Collision detection for continuously deforming bodies. Eurographics, pages 325–333, 2001.

[VMT00] Pascal Volino and Nadia Magnenat-Thalmann. *Virtual Clothing, Theory and Practice*. Springer-Verlag Berlin Heidelberg, 2000.

[VSC01] T. Vassilev, B. Spanlang, and Y. Chrysanthou. Fast cloth animation on walking avatars. *Computer Graphics Forum*, 20, 2001.

[VT95] Pascal Volino and Nadia Magnenat Thalmann. Collision and self-collision detection : Robust and efficient techniques for highly deformable surfaces. *Eurographics Workshop on Animation and Simulation*, 1995.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

BALLET BALANCE STRATEGIES

Camilla Pedersen
IT-University of Copenhagen, Denmark
Kenny Erleben and Jon Sporring*
Department of Computer Science, University of Copenhagen, Denmark

Abstract

Animating physically realistic human characters is challenging, since human observers are highly tuned to recognize human cues such as emotion and gender from motion patterns. Any new effort towards improving the physical realism of animating the human body is therefore valuable both for application and research purposes.

The main contribution of this paper is a new model firmly based on biomechanics. The new model has been developed to animate some basic steps of ballet dancers, and it is supported by computer simulated experiments showing good agreement with biomechanical measurements of real-life dancers.

Keywords: Biomechanics, Balance Strategy, Weight shift strategy, Control Mechanism.

Nomenclature

The following symbols are used in this paper:

τ	Joint torque
$\theta, \Delta\theta$	Joint angle and angular update
k_s, k_d	Spring constancts
$\vec{r}_i$	Position of body part i
$\vec{r}_{cm}, \vec{r}_{cp}$	Position of center of mass and pressure
m_i, M	Mass of body part i and total mass
p_j	Contact point
n_j	Normal force at p_j
$\vec{f}_j$	Contact force at p_j

Introduction

A long term goal of computer graphics is to increase realism and believability into computer generated animations and pictures [1, 31, 18, 5, 37, 4]. With improved rendering techniques, the lack of physical realism and believability is becoming increasingly obvious and annoying to the common observer, and one accompanying long term goal in animation is to increase physical realism by using physics to model plausible behavior and movement of computer models. This known as physics-based animation.

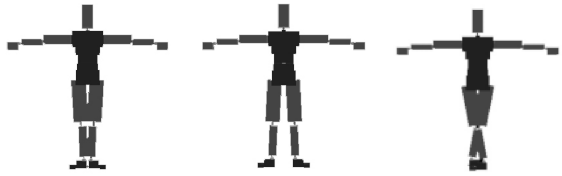
Physics-based animation is a highly interdisciplinary field, based on engineering, physics, and mathematics [42, 36, 3]. Most noticeable is simulation models based on traditional engineering methods used in robotics and construction [9, 14], where forward dynamics is the most popular technique.

In an animated movie it is generally believed that relying 100% on physical principles inhibits the creativity and aesthetics of an animator, and animators typically work using the "Principles of Animation" [25, 15] such as follow through and exaggeration of the motion, to convey the emotions of a character. The implication is that characters may be put in unnatural poses leading to penetrations and cloth tangling.

In recent years the emphasis on physics-based animation have given rise to a new field, "plausible simulation" [6], where new techniques have been proposed, such as sampling the entire range of possible simulations using forward dynamics [19] and optimization of physical constraints [45, 39, 38, 12]. However, animating physically realistic human characters has proven to be very challenging, since human observers are fine tuned to recognize human cues from motion patterns. This is in contrast to the mathematical complexity of simulating natural turbulent phenomena like water and smoke.

This paper studies biomechanical and ballet inspired

*Corresponding author. E-mail: sporring@diku.dk, phone: +45 3532 1469



(a) (b) (c)

Figure 1: Three ballet poses: (a) First, (b) Second, and (c) Fifth. The feet are positioned in the frontal planed in all poses.

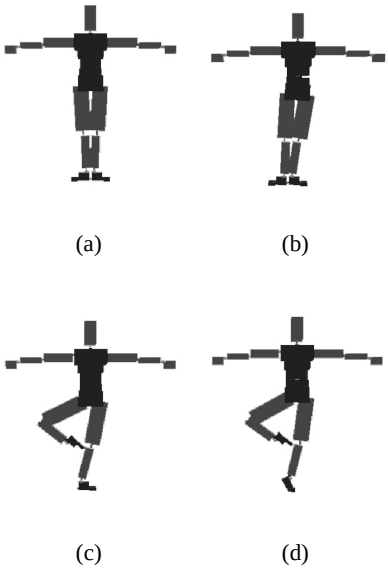
balance and weight shifting strategies. Ballet is a balance art and is a prime focus for learning about human balance and weight shifting strategies. This paper is based on the master thesis by N.N. [35], where an articulated figure is modeled to perform some basic balances and movements of a ballet dancers. N.N. has combined a bachelor in music and dance with ballet dancing and a master degree in computer science.

The articulated figure is initiated from one of the standard poses in quiet standing shown in Figure 1, and the weight shifting strategies used to obtain a quiet standing on one toe will be described. The four sub-goals are shown in Figure 2: Balancing on two legs, weight shift to the supporting leg, balancing on one leg, and balancing on the toe of one leg. The new model is firmly based on biomechanics and is supported by computer simulated experiments showing good agreement with biomechanical measurements of real-life dancers.

Survey of past work

The study of balance has been performed in at least three seperate areas of research: Biomechanics, Robotics, and Animation. We will in the following highligt some research from these areas.

Biomechanics and the study of ballet: Ballet is an art-form, where balance plays a central role. Classical ballet techniques are throughly described in the literature, see e.g. [17, 43]. Biomechanical studies of ballet has mainly been studied through injury cases, e.g. [13]. The most common biomechanical investigations into balancing of the human body



(a) (b) (c) (d)

Figure 2: Shifting the weight from both legs in first pose (a) to the left leg (b), to a one-legged stand (c), and finally to a one-toed (d) quiet standing.

is through the inverted pendulum model [28]. Empirical investigations have shown that the velocity of center of mass plays a role in balancing [44]. Empirical studies on real humans have been performed on balancing of humans versus the position of the center of mass [34]. Weight shifts have been measured using a force-platform [32, 33]. Finally, a thorough measurement of the properties of mass and inertia of human body parts may be found in [10].

Robotics: An early work on balancing robots may be found in [41]. Weight shift strategies for walking are often performed through dynamic walking machines, where there is more or less degree of control involved in the walking cycle [29, 24, 40]. Recently, a number of spectacular and stable humanoid robots have been produced starting with the Honda robot [22]. Especially in relation to our work, the center of pressure has recently been introduced as a stable control mechanism for balancing robots [21, 20].

Animation: A very early mentioning of simulated human motion is given in [2], and early implementations may be found in [16, 30]. The implementation of dancing models is only scarcely discussed in the literature, however one exception is [8]. A major in-

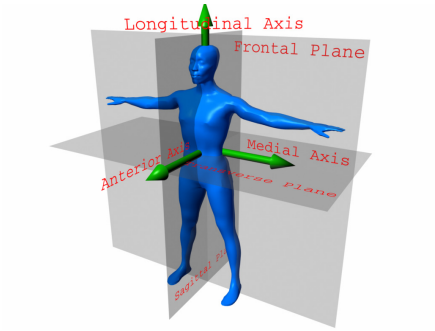


Figure 3: Sagittal, Transverse, and Frontal planes and Medial, Longitudinal and Anterior axis.

spiration for our work has been [26, 46], where center of mass is used to control the balance and motion of a humanoid model of varying complexity. Alternative to center of mass strategies are inverse kinematics e.g. [27], energy models e.g. [7], and learning approaches e.g. [11].

The Biomechanics of Quiet Standing

A human in quiet standing may be modeled by an articulated figure [26, 46] consisting of a set joints and a set of links representing body parts. The set of possible joints consist of revolute (1 Degree of Freedom (DOF)), universal (2 DOF), and ball-and-socket (3 DOF) joints. In this work, we have used Wooten's model [46], which contains 28 DOFs, and uses real measurements of the mass, m_i, center of mass, $\vec{r}_i$, and moments of inertial of all the body parts [10]. The ankle and hip joints is of particular importance for this paper, and they are modeled by an universal joint and a ball-and-socket joint respectively.

Traditionally in biomechanics and anatomy, motion and orientation are described in three planes: The Sagittal (x-axis), the Transverse (y-axis) and the Frontal (z-axis) plane. These planes are illustrated in Figure 3. Measurements of angles and positions are traditionally also performed in these planes [32] by projection onto the respective planes and axes. Typical projections are: The position of the center of mass, the position of the center of pressure, angle of joints, and the direction of gravity. Although it seems needlessly complicated to work with the projections rather than the underlying 3D geometry, it allows for comparison with the substantial biomechanical literature.

Muscles are used to move and sustain posture of the human skeleton, and our articulated figure is supplemented by an actuator system, which applies joint torques according to a simple damped angular spring model,

$$\tau = k_s^{\text{muscle}} \left(\theta^{\text{target}} - \theta^{\text{current}} \right) - k_d^{\text{muscle}} \dot{\theta}^{\text{current}}. \quad (1)$$

In the equation, τ is the length of the torque vector, θ^{target} and θ^{current} are target and current angles, $\dot{\theta}^{\text{current}}$ is the current velocity of the angle, and k_s^{muscle} and k_d^{muscle} are spring and damping constants.

A balance control strategy is a function that determine updates, $\Delta\theta$'s, based on the current state of the articulated figure, θ^{current}'s, such that the model will converge towards a desired state of quiet balance. I.e. the strategy iteratively determines new parameter values, θ^{new}'s as,

$$\theta^{\text{new}} = \theta^{\text{current}} + \Delta\theta. \quad (2)$$

In the rest of this article, it will be assumed that the model is placed on a planar floor, and the contact between the floor and the feet is represented by a set of coplanar contact points, $\vec{p}_j$. The support polygon is defined as the 2D convex hull of all the contact points. The center of mass is of the human model is defined as

$$\vec{r}_{\text{cm}} = \frac{1}{M} \sum_i^N m_i \vec{r}_i, \quad (3)$$

where $M = \sum_i^N m_i$ is the total mass, N is the number of body parts in the model, m_i is the individual weights of the body parts, and r_i are their locations. Note that center of mass is not fixed w.r.t. any location of the body during motion of the individual body parts. The center of pressure is defined as

$$\vec{r}_{\text{cp}} = \frac{1}{||\vec{n}||} \sum_j^K ||\vec{n}_j|| \vec{p}_j, \quad (4)$$

where $\vec{n} = \sum_j^K \vec{n}_j$ is the total normal force acting on the human model, $\vec{n}_j$ is the normal force applied to the human model at the j'th contact point $\vec{p}_j$, and K is the number of contact points.

For simplicity, a spring model for the floor contact forces is used, where the contact force of the j'th contact is

$$\vec{f}_j = k_s^{\text{contact}} \left(\vec{p}_j^{\,\text{initial}} - \vec{p}_j \right) - k_d^{\text{contact}} \dot{\vec{p}}_j. \quad (5)$$

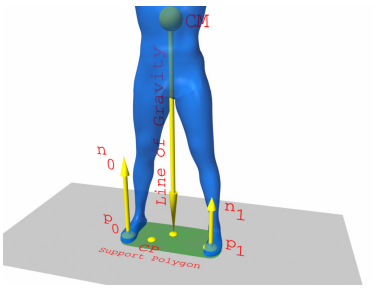


Figure 4: Illustration of biomechanical definitions: Center of mass, center of pressure, support polygon and line of gravity

In the equation, $\vec{p}_j^{\,\text{initial}}$ is the initial point of contact, $\vec{p}_j$ is the current contact point, $\dot{\vec{p}}_j$ is the velocity of the current contact point, and k_s^{contact} and k_d^{contact} are spring and damping constants. The vector, $\vec{n}_j$, is calculated as the projection of $\vec{f}_j$ onto the contact normal of the floor, and the tangential part is a simple model of frictional force.

Contact forces can only be repulsive, and attractive contact forces, f_j, are therefore set to zero. Slipping is obtained by setting $\vec{p}_j^{\,\text{initial}}$ equal to $\vec{p}_j$, when the magnitude of the tangential force component exceed a multiple of the magnitude of the normal force component, $||\vec{f}_{\text{friction}}|| \leq -\mu||\vec{n}||$.

The line of gravity is defined as the line going from the center of mass to the ground in the direction of the gravitational field. The point of intersection between the line of gravity and the floor is referred to as the projection of the center of mass. These concepts are illustrated in Figure 4.

Balance is defined as an objects ability to maintain quiet standing, where quiet standing is the state, where the projection of the center of mass is kept within the support polygon [23, 26, 46]. The implication is that the greater support polygon, the lower center of mass, the more stable the balance and vice versa. The human body has a highly placed center of mass over a rather small support polygon, and as such the human body behaves as an inverted pendulum.

In the remainder of this paper, control strategies for maintaining quiet standing and moving an articulated figure from one pose of quiet standing to another will be discussed.

Balance Strategies: Mass Center versus Pressure Center

In the following, two strategies for balance will be compared: The center of mass strategy and the center of pressure strategy. The analysis are done on an articulated figure based on [46] standing with parallel feet. The balance is controlled by the ankle and hip joint and only in Sagittal plane in agreement with [44].

The center of mass strategy is the traditional balance strategy, where the angular change is controlled as a function of the projection of the center of mass onto the support plane [26, 46]:

$$\Delta\theta = k_s^{\text{cm}}\left(r_{\text{cm}}^{\text{current}} - r_{\text{cm}}^{\text{target}}\right) - k_d^{\text{cm}} v_{\text{cm}}, \quad (6)$$

where $\Delta\theta$ is the angular change of the ankle in the Sagittal plane, $r_{\text{cm}}^{\text{current}}$ and $r_{\text{cm}}^{\text{target}}$ are the projections of the current and target positions of the center of mass onto the anterior axis, v_{cm}, is the velocity vector of the center of mass projected onto the anterior axis, and k_s^{cm} and k_d^{cm} are control parameters.

In the center of pressure strategy, the model uses the center of pressure to control the center of mass, and when the positions are right above each other the pendulum is in perfect balance. The goal of this strategy is therefore to calculate a desired position of the center of pressure, and use this for controlling the muscles.

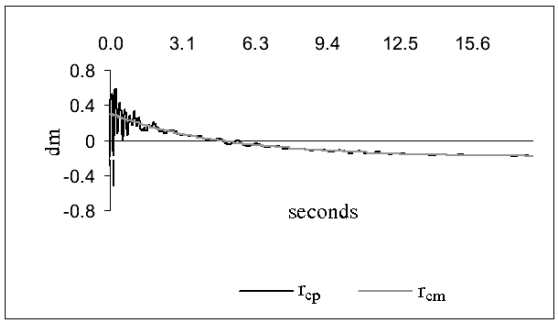
$$\Delta r_{\text{cp}} = k_s^{\text{cm}}\left(r_{\text{cm}}^{\text{current}} - r_{\text{cm}}^{\text{target}}\right) - k_d^{\text{cm}} v_{\text{cm}}, \quad (7)$$

$$r_{\text{cp}}^{\text{target}} = \Delta r_{\text{cp}} + r_{\text{cm}}, \quad (8)$$

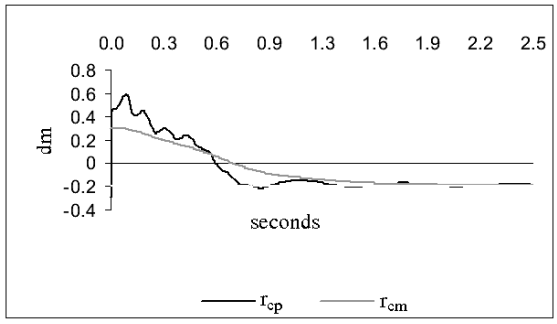
$$\Delta\theta = k_s^{\text{cp}}\left(r_{\text{cp}}^{\text{current}} - r_{\text{cp}}^{\text{target}}\right) - k_d^{\text{cp}} v_{\text{cp}}, \quad (9)$$

where Δr_{cp} is the positional change of the projection of the center of pressure onto the anterior axis, $r_{\text{cp}}^{\text{current}}$ and $r_{\text{cp}}^{\text{target}}$ are the projections of the current and target positions of the center of projection onto the anterior axis, and k_s^{cp} and k_d^{cp} are control parameters.

In Figure 5 the dynamics of the projected center of mass and the center of pressure on the anterior axis is shown. The model manages to balance using both strategies, but the center of pressure strategy requires only approximately 1.5 sec., while the center of mass strategy requires almost 10 sec.. In addition, the center of pressure strategy also has the smallest amplitude of the oscillations of the center of pressure, which means that it has the best control over the contact with the ground.



(a)



(b)

Figure 5: Comparison of the dynamics of the ankle for the two strategies: (a) Center of mass strategy and (b) Center of pressure strategy. Motion is restricted to the Sagittal plane.

Dynamics of a Ballet Dancer

Quiet standing on the toe of one leg is central in all ballet training. This is demanding, since the dancer has to balance on a very small support polygon while at the same time looking at ease. Both in quiet balance and in most of the basic ballet exercises, the legs are strictly separated into the working leg (doing the exercise) and the supporting leg. Shifting the weight between the legs is therefore an important task, and it should preferably be done without drawing the attention of the audience. In [35], strategies for obtaining a balance on the left leg with the right foot by the left knee were developed as illustrated in Figure 2. This pose is used, when dancers turn in a pirouette, and it is perhaps the most basic of all balances in ballet training.

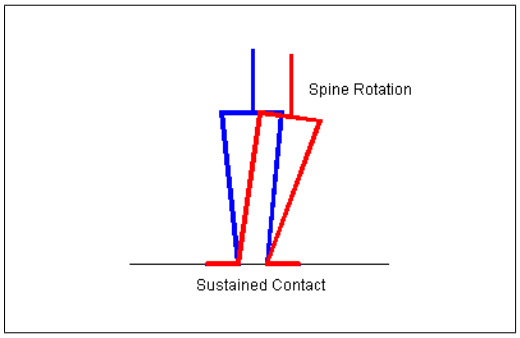


Figure 6: A rotation in the ankles with straight legs implies a rotation in the hip, and a rotation in the spine is required in order to keep the upper body vertical.

Weight Shifting Strategies

The weight shifting strategy described in the following was inspired by the analysis of real dancers presented in the literature [32], where it was shown that changes in angles in the hips and ankles are nearly identical during weight shift, and that the center of pressure start moving towards the working leg and end up being on the supporting leg. The movement of the center of pressure was tested on the articulated figure for the shifting of the weight in the frontal plane from a position between the feet to the left foot as shown in Figures 2(a)-2(b), and the center of pressure strategy showed good agreement with the measurements on real dancers.

To shift the weight, the left ankle was designated to be the controlling joint using (9), while the remaining joints were used to keep balance and to control the position of the upper body. Keeping the right foot flat on the ground, the angular changes in the right ankle and the hips are calculated from the angular change in the left ankle.

Ballet aesthetics requires that the upper body is kept parallel to the line of gravity during a weight shift in the frontal plane with both feet fixed on the floor. It has been claimed [32] that dancers keep their upper body vertical by a counter rotation in the hip joints, however this is only physically possible when the legs are parallel as illustrated in Figure 6. Ballet dancers compensate for the hip rotation by a counter rotation in the lumbar region of the lower back [17], and therefore a control function in the spine is required. Dancers control the body center by the stomach muscles, and experience [35] has shown that

these stomach muscles are extremely important for aesthetic motion of the articulated figure, thus a control function in the pelvis is used to inhibit rotation in the Sagittal plane. Both control functions are modeled using a spring law similar to (1).

The final weight shift of the articulated figure is shown in Figure 2. The resulting articulated figure agrees with measurements performed on real ballet dances [32] as follows: The measured angles in quiet standing are identical, and at initial weight shift, the center of pressure starts to move towards the right foot and ends in a position on the left foot as seen in Figure 5(b).

Movement to Quiet Standing on One Toe

The final movements to obtain a one-toed quiet standing is achieved by lifting the non-supporting leg, and shifting from a foot stand to a toe stand. See Figure 2. It is not difficult to raise the leg, however the major challenge is to keep balance on a very small support polygon.

Two strategies have been developed: A strategy for raising the right leg, and a strategy for making a weight shift to the toe. Similarly to the weight shifting balance strategy, both are based on the center of pressure strategy.

To raise the leg, the left ankle is kept as the controlling joint, and the center of pressure strategy is used to keep the balance on the center of the left foot. To shift the weight to one toe, the center of pressure strategy is used in two steps: Firstly, for the controlling ankle joint, and secondly to control the toe joint, when the ankle has been straightened.

Lifting the right leg to an aesthetically pleasing pose is performed using a spring law. Both the lifting of the leg and the shifting of the weight to the toe produces a motion of the center of mass, and the left hip is used to maintain balance by (6). The spine and pelvis are controlled as explained in the previous section. The analysis of the results shows a motion of the center of pressure, which is not entirely in agreement with measurements on real dances [32].

Discussion

This paper has compared two strategies for an articulated figure in quiet standing: Center of mass and center of pressure. It has been demonstrated that controlling balance using the center of pressure is the fastest strategy and is also in best agreement with biomechanical measurements on ballet dancers. From a ballet point of view, controlling the center of pressure is a way of controlling the contact of the feet with the ground. Ballet-dancers are very much aware of the relation between their feet and the ground, since it strongly influences their balance, their stance, and the audience's impression of the dancer's body. A ballet dancer must have a light impression.

Further in contrast to some speculations in the literature, the model presented in this paper has shown that it is most natural to use the ankles and not the hip joints for balance control.

Using spring laws for modeling actuator forces and contact forces are attractive due to their simplicity although they require a lot of parameter tuning in practice.

Balance and weight shifting are the most basic techniques learnt in ballet. Future steps in our research will be to develop strategies for exercises on one leg. However, it remains to be demonstrated that a small collection of motion strategies are sufficient to simulate the physical motion of human beings. It is our belief that this is the case.

References

[1] William W. Armstrong and Mark W. Green. The dynamics of articulated rigid bodies for purposes of animation. *The Visual Computer*, 1(4):231–240, 1985.

[2] Norman I. Badler and Stephen W. Smoliar. Digital representations of human movement. *Computing Surveys*, 11(1), 1979.

[3] David Baraff. Physical based modeling: Rigid body simulation. ONLINE SIGGRAPH 2001 COURSE NOTES, Pixar Animation Studios, 2001. http://www-2.cs.cmu.edu/ baraff/sigcourse/.

[4] B.A. Barsky, N. Badler, and D. Zeltzer, editors. *Making Them Move: Mechanics Control and Animation of Articulated Figures*. The Morgan Kaufmann Series in Computer Graphics and Geometric Modeling. Morgan Kaufman Publishers, Inc., 1991.

[5] R. Barzel and A.H. Barr. A modeling system based on dynamic constraints. In *Computer Graphics*, volume 22, pages 179–187, 1988.

[6] Ronen Barzel, John F. Hughes, and Daniel N. Wood. Plausible motion simulation for computer graphics animation. In *Proceedings of the Eurographics Workshop, Computer Animation and Simulation*, pages 183–197, 1996.

[7] Ronan Boulic, Ramon Mas-Sanso, and Daniel Thalmann. Complex character positioning based on a compatible flow model of multiple supports. *IEEE Transactions on visualization and computer graphics*, 3(3), 1997.

[8] Tom Calvert. Composition of realistic animation sequences for multiple human figures. In Barsky et al. [4], chapter 2.

[9] John J. Craig. *Introduction to Robotics, mechanics and control*. Addision-Wesley Publishing Company, Inc, second edition edition, 1986.

[10] Wilfred Taylor Dempster and George R. L. Gaughran. Properties of body segments based on size and weight. *The American Journal of Anatomy*, 1967.

[11] Petros Faloutsos, Michiel van de Panne, and Demetri Terzopoulos. Composable controllers fro physics-based character animation. In Eugene Fiume, editor, *SIGGRAPH 2001, Computer Graphics Proceedings*, pages 251–260. ACM Press / ACM SIGGRAPH, 2001.

[12] Anthony C. Fang and Nancy S. Pollard. Efficient synthesis of physically valid human motion. *ACM Transactions on Graphics (TOG)*, 22(3):417–426, 2003.

[13] Donald F. Featherstone. *Dancing without danger*. Kaye and Ward Limited, England, 1970.

[14] Roy Featherstone. *Robot Dynamics Algorithms*. Kluwer Academic Publishers, second printing edition, 1998.

[15] Ollie Frank and Johnson Thomas. *Illusion of Life: Disney Animation*. Hyperion Press, 1995.

[16] M. Girard and A. A. Maciejewski. Computational modeling for the computer animation of legged figures. In *ACM Siggraph 1985*, pages 263–270, 1985.

[17] Valerie Grieg. *Inside Ballet Technique*. Dance Books, London, UK, 1994.

[18] J. K. Hahn. Realistic animation of rigid bodies. In *Computer Graphics*, volume 22, pages 299–308, 1988.

[19] John Hughes and Ronen Barzel. Siggraph 2003 course, plausible simulation. SIGGRAPH 2003 Conference Select CD-ROM and "Full Conference DVD, 2003.

[20] Satoshi ITO, Hironori ASANO, and Haruhisa KAWASAKI. A balance control in biped double support phase based on center of pressure of ground reaction forces. In *The 7-th IFAC Symposium on Robot Control*, volume 1, pages 205–210, Wroclaw, 2003.

[21] Satoshi Ito and Haruhisa Kawasaki. A standing posture control based on ground reaction force. In *Proceedings of the 2000 IEEE/RSJ International Conference on Intellgent Robots and Systems*, pages 1340–1345, Takamatsu, 2000.

[22] Fumio Kanehiro, Kenji Kaneko, Kiyoshi Fujiwara, Kensuke Harada, Shuuji Kajita, Kazuhito Yokoi, Hirohisa Hirukawa, Kazuhiko Akachi, and Takakatsu Isozumi. The first humanoid robot that has the same size as a human and that can lie down and get up. In *Proceedings of the 2003 IEEE International Conference on Robotics & Automation*, pages 1633–1639, Taipei, Taiwan, September 2003.

[23] Ellen Kreighbaum and Katharine M. Barthels. *Biomechanics, a qualitative approach for studying human movement*. Allyn and Bacon, Toronto, Canada, 4th edition, 1996.

[24] Andrew L. Kun and W. Thomas Miller. Adaptive dynamic balance of a biped robot using neural networks. In *Proc. IEEE Int. Conf. on RA*, pages 240–245, 1996.

[25] John Lasseter. Principles of traditional animation applied to 3d computer animation. In *Proceedings of the 14th annual conference on*

Computer graphics and interactive techniques, pages 35–44. ACM Press, 1987.

[26] Joseph Laszlo. Controlling bipedal locomotion for computer animation. Master's thesis, University of Toronto, Canada, 1996.

[27] Joseph Laszlo, Michiel van de Panne, and Eugene Fiume. Limit cycle control and its application to the animation of balancing and walking. In *Proceedings of SIGGRAPH 1996*, pages 155–162, New Orleans, LA, 1996.

[28] Column D. Mackinnon and David Winter. Control of whole body balance in the frontal plane during human walking. *Journal of Biomechanics*, 26(6):633–644, 1998.

[29] Tad McGeer. Passive dynamic walking. *The International Journal of Robotics Research*, 9(2):62–82, 1990.

[30] Michael McKenna and David Zeltzer. Dynamic simulation of autonomous legged locomotion. *Computer Graphics*, 24(4):29–38, 1990.

[31] M. Moore and J. Wilhelms. Collision detection and response for computer animation. In *Computer Graphics*, volume 22, pages 289–298, 1988.

[32] L. Mouchnino, R. Aurenty, J. Massion, and A. Pedotti. Coordination between equilibrium and head-trunk orientation during leg movement: A new strategy built up by training. *Journal of Neurophysiology*, 67(6), 1992.

[33] E. Otten. Balancing on a narrow ridge: biomechanics and control. *Philisophical Transactions: Biological Sciences*, 354(1385):869–875, 1999.

[34] Yi-Chung Pai and James Patton. Center of mass velocity-position predictions for balance control. *Journal of Biomechanics*, 30(4):347–354, 1997.

[35] Camilla Pedersen. Balancekontrol og strategier til en dynamisk animation af en balletdanser [balance control and strategies for a dynamical animation of a ballet dancer]. Master's thesis, IT-University of Copenhagen, Denmark, 2002.

[36] Friedrich Pfeiffer and Christoph Glocker. *Multibody dynamics with unilateral contacts*. Wiley series in nonlinear science. John Wiley & Sons, inc., 1996.

[37] J.C. Platt and A.H. Barr. Constraint methods for flexible bodies. In *Computer Graphics*, volume 22, pages 279–288, 1988.

[38] Jovan Popovic, Steven M. Seitz, and Michael Erdmann. Motion sketching for control of rigid-body simulations. *ACM Transactions on Graphics*, 22(4):1034–1054, 2003.

[39] Jovan Popović, Steven M. Seitz, Michael Erdmann, Zoran Popović, and Andrew Witkin. Interactive manipulation of rigid body simulations. *Proceedings of SIGGRAPH 2000*, pages 209–218, July 2000. ISBN 1-58113-208-5.

[40] Jerry E. Pratt and Gill A. Pratt. Exploiting natural dynamics in the control of a 3D bipedal walking simulation. In *International Conference on Climbin and Walking Robots (CLAWAR9i9)*, Portsmouth, UK, 1999.

[41] M.H. Raibert. *Legged Robots that balance*. MIT Press, Cambridge, MA, 1986.

[42] D.E. Stewart and J.C. Trinkle. An implicit time-stepping scheme for rigid body dynamics with inelastic collisions and coulomb friction. *International Journal of Numerical Methods in Engineering*, 1996.

[43] Gretchen Warren. *Classical Ballet Technique*. University Press of Florida, 1989.

[44] David Winter, Aftab Patla, Francois Prince, Milad Ishac, and Krystyna Gielo-Perczak. Stiffness control of balance in quiet standing. *Journal of Neurophysiology*, 80(3):1211–1221, 1998.

[45] Andrew Witkin and Michael Kass. Spacetime constraints. *ACM SIGGRAPH, Computer Graphics*, 22(4):159–168, 1988.

[46] Wayne L. Wooten. *Simulation of Leaping, Tumbling, Landing, and Balancing Humans*. PhD thesis, Georgia Institute of Technology, USA, 1998.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

REAL-TIME SIMULATION OF SMOKE USING GRAPHICS HARDWARE

Marinus Rørbech*
Department of Computer Science
University of Copenhagen
Denmark

Abstract

Based on earlier presented solvers for the incompressible Navier-Stokes equations, we implement a 3D fluid solver. The solver works in a few simple steps implemented in fragment shaders running entirely on the *Graphic Processing Unit* (GPU) of a modern graphics card. We present a simple visualization approach for rendering high-quality smoke animations in real-time. Finally, to handle interaction between fluid and stationary objects, we present a simple method for setting boundary conditions.

Keywords: Fluid dynamics, Navier-Stokes, Real-time, GPU.

Nomenclature

d	Density field
$\mathbf{f}$	Force field
h	Spatial discretization
$\mathbf{N}$	Vorticity location field
p	Pressure field
T	Temperature field
T_0	Mean temperature
t	Time
$\mathbf{u}$	Velocity field
β	Thermal expansion coefficient
ε	Vorticity confinement control factor
η	Gradient field of vorticity magnitude
ω	Vorticity
∇	Gradient operator $\nabla = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z})$
ν	Kinematic viscosity
ρ	Density

Introduction

With the increase of computational power in standard home computers, more and more physical simulation is put into computer games. Recently, a quantum leap in computational power has been made with the introduction of the programmable GPU, available on modern graphics adapters. This has lead to higher quality in game graphics, and developers have started to use the GPU for non-graphics computations. This leap will believably open for more advanced effects in upcoming game titles.

Over the past years, a lot of the focus in game development has been dedicated to the simulation of rigid body physics. We believe that one of the new features which will show up in computer games in near future is the simulation of fluid dynamics.

To encourage this, we describe a simple method for achieving real-time animated 3D smoke.

Previous Work

The motion of a non-turbulent, viscous fluid is modeled by the *incompressible Navier-Stokes equations*

$$\frac{\partial \mathbf{u}}{\partial t} = -(\mathbf{u}\cdot\nabla)\mathbf{u} + \nu\nabla^2\mathbf{u} - \frac{1}{\rho}\nabla p + \mathbf{f} \quad (1)$$

$$\nabla\cdot\mathbf{u} = 0 \quad (2)$$

where (1) describe the change in the fluid velocity, and (2) enforces conservation of mass.

Numerical methods for solving these equations have been researched extensively during the last decade, and today very realistic fluid animations can be generated off-line. Recent years, efforts have been made to take the methods for solving these equations to the next level: Real-time simulation.

One of the first methods for simulating the full 3D Navier-Stokes equations for the purpose of Computer Graphics, is presented in [4]. The terms of the equations are solved step by step, using a first order

central differencing approach. The velocity is made divergence-free, by solving the pressure term iteratively, using a *Successive Over-Relaxation* (SOR) method [2]. Because the solver in [4] is based on explicit methods the solver is only stable for small time-steps.

In [8], the first step towards real-time simulation is taken by making the solver unconditionally stable. The advection term, $(\mathbf{u}\cdot\nabla)\mathbf{u}$, is solved using a *semi-Lagrangian* integration scheme, where points are traced backwards to find advected velocities. Diffusion, described by the term $\nu\nabla^2\mathbf{u}$, is solved implicitly using a multigrid method [1]. To remove divergence from the velocity field, a pressure field is computed by solving the Poisson equation

$$\nabla^2 p = -\nabla\cdot\mathbf{u} \tag{3}$$

also using a multigrid method. The pressure gradient is then used to adjust the velocity field $\mathbf{u}$ by

$$\tilde{\mathbf{u}} = \mathbf{u} - \nabla p \tag{4}$$

producing the divergence-free velocity field $\tilde{\mathbf{u}}$.

In [3], the method from [8] is refined for the special case of simulating smoke. The diffusion term is neglected based on the assumption that the effect of diffusion is damped out, because of the low viscosity of air and the coarseness of the simulation grid. To make up for some of the small-scale detail thus missing, a vorticity confinement force [10]

$$\mathbf{f}_{vc} = h\,\varepsilon(\mathbf{N}\times\omega) \tag{5}$$

is introduced, where

$$\omega = \nabla\times\mathbf{u} \tag{6}$$

and

$$\mathbf{N} = \frac{\eta}{|\eta|}, \quad \eta = \nabla|\omega| \tag{7}$$

This adds a swirling behavior, which looks very natural for smoke. Derived from [5] a simple thermal buoyancy force, given by

$$\mathbf{f}_T = \beta\,(T - T_0)\cdot(0,1,0)^T \tag{8}$$

is also added, enabling various temperature effects, such as radiators causing the air to rise.

In [9], a simple, easy implementable solver is presented, based on the method from [8]. The multigrid method is replaced with the much simpler Gauss-Seidel method [1]. The entire solver consists of approximately 100 lines of C-code, and is thus presented directly in the paper.

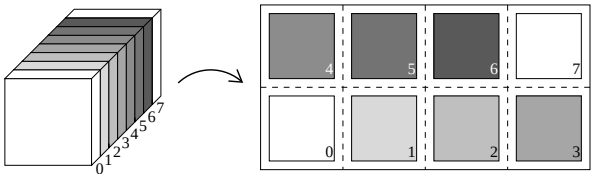


Figure 1: Flat 3D texture – the slices of a 3D field is laid out as tiles in a 2D texture. This figure is copied from [6].

Although this method is very efficient, software implementations are still not quite fast enough for real-time simulations in 3D. In [6], a method for simulating clouds is implemented using fragment shaders, running entirely on a GPU. To do this, 3D vector fields are represented as *flat 3D textures*, see Figure 1. Because of the inability of fragment shaders to read and write the same memory, the Gauss-Seidel approach of [9] is replaced with a variant of the Jacobi method, called *Red-Black Gauss-Seidel* [1]. Due to the power of the GPU, this simulation runs at interactive rates.

The Method

As in [8], we solve the Navier-Stokes equations in two steps. The first step updates the velocity field according to (1). This leads to a non-conserving velocity field. In the second step, we remove divergence, producing a mass-conserving velocity field upholding (2).

Updating Velocity

Advection of the velocity is solved using the semi-Lagrangian advection scheme presented in [8]. This method is stable for large time steps, which is crucial to our real-time simulation.

Following the method from [3] we neglect the diffusion term. Since the kinematic viscosity of air is very small, the effects of diffusion are assumably damped out by the numerical dissipation of our coarse grid.

Having left out the diffusion term, we adopt the vorticity confinement force, described in [3]. This

makes up for the small-scale detail in the velocity field that is dissipated by our coarse grid, and adds natural looking swirls to our smoke.
To easily interact with the smoke, we also include the addition of an external force field, which can respond to any user action.
Finally, we add a thermal buoyancy force, as described in [3], to be able to simulate various temperature-related effects.

Removing Divergence

As in [9], the divergence is removed in three simple steps. First the divergence is computed. Then we compute the pressure field by solving (3) iteratively. Following [6], we use the simple iterative Jacobi method [1], which, opposed to the Gauss-Seidel method used in [9], is implementable on a GPU. Finally, the velocity field is adjusted according to (4).

Implementation on GPU

The GPU is a programmable processor available on most recent graphics adapters. Even though it is designed for graphics processing, general purpose processing on a GPU is attractive due to its computational power.
The GPU works on streams of vertices and fragments (pixels with attributes, such as depth and color), and has only memory read access through texture maps. Thus, to implement a fluid solver to run on a GPU, the fluid state should be represented with texture maps, and the solver steps should be implemented as *fragment shaders*.

Vector Field Representation

We adopt the vector field representation described in [6]. A 3D vector field is represented by laying out each slice as a tile in a 2D texture, as can be seen in Figure 1. This allows us to update the entire field in a single render pass.

Fragment Shaders

The fragment shaders are executed by drawing regular primitives, such as triangles or quadrilaterals. When these primitives are rasterized, the fragment shaders are executed once on each resulting fragment, and the output is written to the frame buffer.

The input to a fragment shader, in the context of our fluid solver, is a texture coordinate referring to a specific cell in the simulation grid. The value of any field in this point can be looked up using a texture read instruction on the respective field texture map.

For looking up neighbor values, offsets can be added to the input texture coordinate to produce texture coordinates representing other cells in the grid. Neighbors in x and y directions can easily be found by adding the vectors $(1,0,0)^T$, $(1-,0,0)^T$, $(0,1,0)^T$, and $(0,-1,0)^T$, respectively, to the texture coordinate of the current cell. For neighbors in different tiles, however, offsets to the respective tiles must be taken into account. As opposed to [6], we pre-compute relative offsets between tiles, and pass them to our fragment shaders as texture coordinates. Compared to [6] this saves us a texture-read instruction per grid cell per simulation step (approximately 10^6 texture instructions per frame at resolution 30x30x30).

For each solver step we have a fragment shader which processes only a single cell in the grid with respect to that solver step. The entire grid is then processed by drawing primitives covering all the *interior cells* in the flat 3D texture, corresponding to the shaded areas in Figure 1. The *boundary cells* (white areas in Figure 1) are processed separately with special fragment shaders, to enforce the desired boundary conditions of the system.

Visualization

One of the hurdles of simulating fluid in real-time applications is the visual presentation of the fluid. A popular method for rendering smoke in games is to use a particle system, and render each particle using a bill-board texture, see Figure 2. Small scale effects, such as exhaust from a car, can be animated convincingly using a limited number of particles (< 500). However, the particle method is not very scalable since the simulation time increases linearly with the number of particles.

Instead, for animating the smoke we use a density field, as proposed in [9]. The density should simply be moved along with the velocity, so we model the

Figure 2: Bill-boarding – the particles to the left are rendered using the texture mapped quadrilateral in the center, yielding the smoke- or cloud-like appearance to the right.

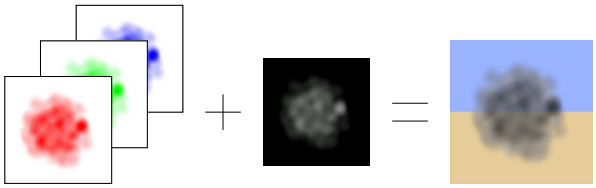


Figure 3: The density texture represents four density fields of red, green, blue (left), and alpha densities (center). When rendered, these four density fields represent colored, transparent smoke (right).

density movement by

$$\frac{\partial d}{\partial t} = (\mathbf{u} \cdot \nabla) d \tag{9}$$

which is similar to the advective term of (1). Thus, we update the position of the density by using the same semi-Lagrangian advection scheme, as described for updating fluid velocity.

Similar to all other vector fields, we represent our density field in a flat 3D texture. This allows us to easily represent colored and transparent smoke, by using the standard RGB color model and the alpha channel of the texture. This actually corresponds to having four density fields, (d_r, d_g, d_b, d_a), representing the red, green, blue, and alpha channels, respectively, see Figure 3. Since fragment operations are vector based, we can update all four density fields at the same speed as updating a single density layer.

The flat 3D texture representation of the colored density field allows us to easily render the density field by simply rendering each slice of the field as a bill-board. This involves drawing a texture mapped quadrilateral per slice in the texture. Slices are drawn in order of distance to the viewer, starting from the back. Each slice is blended with the background, according to density color and alpha value, giving a transparent look. This method is extremely fast, thus, allowing high-rate animations.

To avoid using expensive volume rendering techniques, for instance as presented in [7], we only use 16-bit fixed-point precision in the density texture. In this way, the density texture can be automatically filtered by the graphics adapter, improving image quality and speeding up the updating process.

By representing smoke with a density field, the animation speed is independent of the amount of smoke in the simulation. Unfortunately, this also means that the resolution of the smoke is limited in the same way as the resolution of the simulation grid.

Interaction With Objects

To extend our smoke simulation, we present a simple way to simulate interaction with stationary objects. We represent voxelized objects in a flat 3D texture. In this *obstacle map* we represent cells occupied by an object with the value 0, and cells occupied by the fluid with the value 1. Using a fragment shader, values in any field can easily be masked with regards to the objects in the fluid, allowing simple boundary conditions. Thus, simple stationary objects can be modeled on cell-level.

Results

In this section we show some examples of smoke simulated with our solver. All examples are run on an ATI Radeon 9600 PRO graphics adapter. The grid resolution is $30 \times 60 \times 30$ grid, and the obtainable frame rate is approximately 30 fps. The animations are available for downloading at `http://www.roerbech.dk/marinus/sims.html`.

First, we show a simple smoke stream, which rises due to thermal buoyancy and an external vent force. Frames from this animation can be see in Figure 4.

In the second example, we add a spherical obstacle into the scene from Example 1, to show interaction with stationary objects. Frames from this animation can be seen in Figure 5.

Finally, to demonstrate the capabilities of our colored density field, Figure 6 shows frames from an animation using two different colors of smoke.

Conclusion

We have presented a fast way of animating 3D smoke based on simulation of fluid dynamics.
We keep in mind, that our solver is not yet fully optimized, and we estimate a possible improvement in performance of about 10–15% on this account alone. Further more, the presented examples have been run on an ATI Radeon 9600 PRO graphics adapter, which is already considered obsolete by current graphics hardware standards.
Thus, even though our solver has some limits, it gives the impression that real-time fluid effects in games should be possible in the very near future. Especially when taking into account the extended features for general purpose processing presented on next-generation graphics adapters.
Since our solver is based on the Navier-Stokes it can be used for other purposes than animating smoke, for instance the simulation and animation of explosions, animation of miscellaneous wind effects, such as dry leafs swirling along the ground, etc.

References

[1] Demmel JW. *Applied Numerical Linear Algebra*. Philadelphia: Society for Industrial and Applied Mathematics (SIAM), 1997.

[2] Faires JD, Burden RL. *Numerical Analysis*. 6th edition. Brooks/Cole Publishing Company, 1997.

[3] Fedkiw R, Stam J, Jensen HW. *Visual simulation of smoke*. In: Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, pages 15–22. ACM Press, 2001.

[4] Foster N, Metaxas D. *Realistic Animation of Liquids*. Graphical models and image processing: GMIP 1996, 58(5):471–483.

[5] Foster N, Metaxas D. *Modeling the motion of a hot, turbulent gas*. Computer Graphics 1997, 31(Annual Conference Series):181–188.

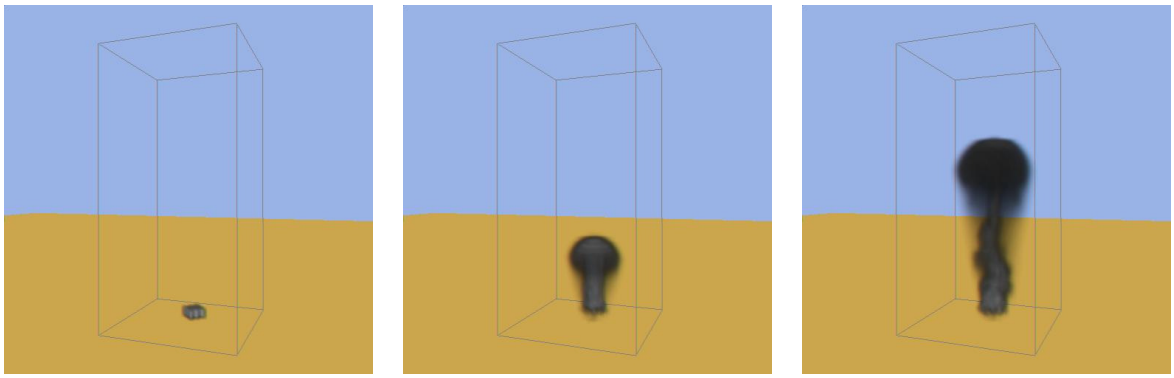
[6] Harris MJ, Baxter WV, Scheuermann T, Lastra A. *Simulation of Cloud Dynamics on Graphics Hardware*. In: Proceedings of the ACM SIGGRAPH/EUROGRAPHICS Conference on Graphics Hardware, Eurographics Association, 2003

[7] Meissner M, Guthe S. *Interactive Lighting Models and Pre-Integration for Volume Rendering on PC Graphics Accelerators*. In: Graphics Interface Proceedings, pages 209–218. Canadian Information Processing Society, 2002.

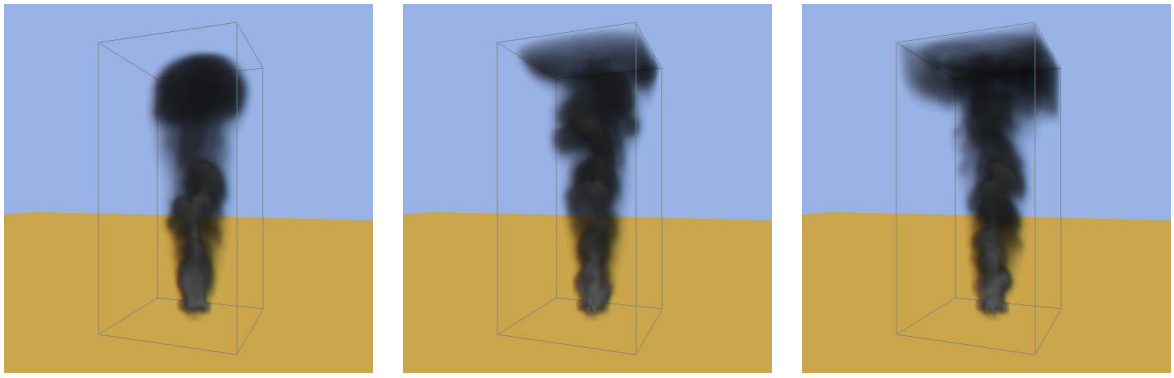
[8] Stam J. *Stable Fluids*. In: Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques, pages 121–128. ACM Press/Addison-Wesley Publishing Co., 1999.

[9] Stam J. *Real-Time Simulation of Fluid Dynamics for Games*, 2003, Available at `http://www.dgp.utoronto.ca/~stam/reality/Research/pub.html`.

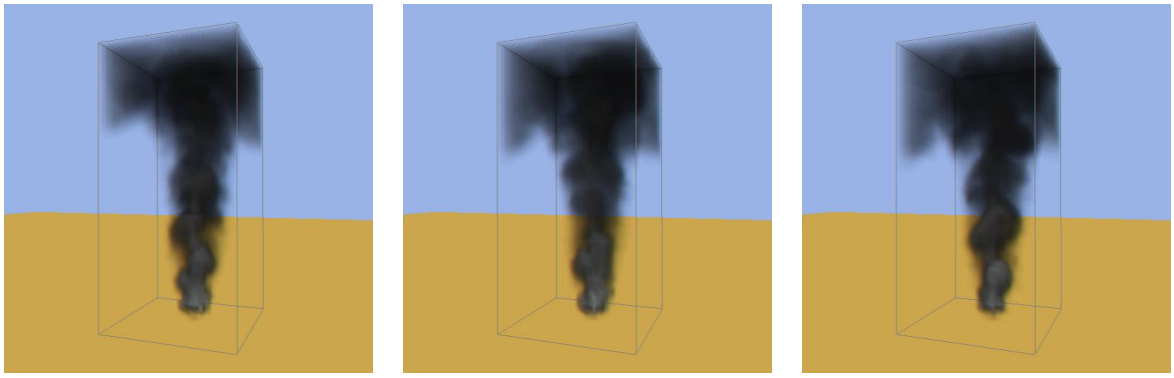
[10] Steinhoff J, Underhill D. *Modification of the euler equations for "vorticity confinement": Application to the computation of interacting vortex rings*. Physics of Fluid 1994, 6(8):2738–2744.



$t = 0.0$ sec. $t = 1.0$ sec. $t = 2.0$ sec.

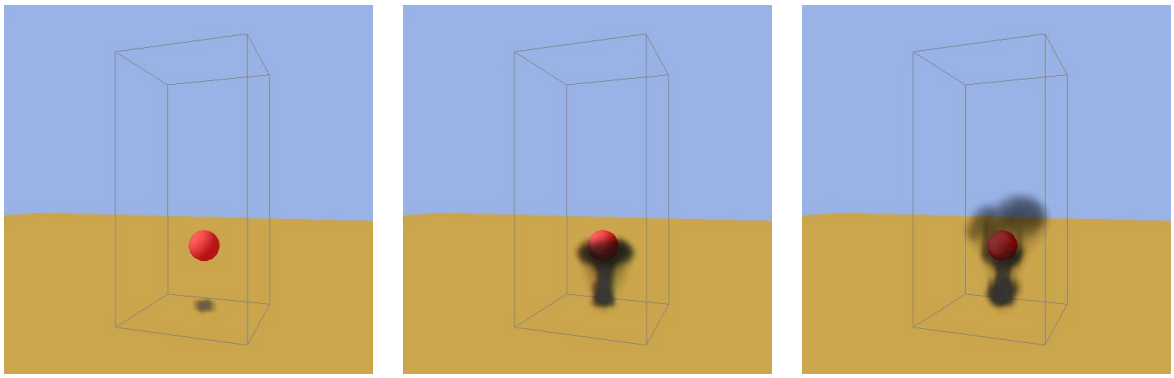


$t = 3.0$ sec. $t = 4.0$ sec. $t = 5.0$ sec.

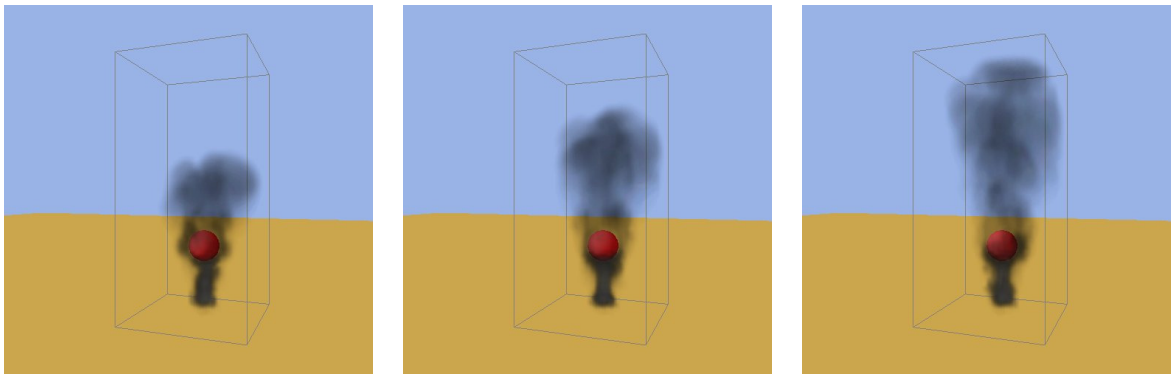


$t = 6.0$ sec. $t = 7.0$ sec. $t = 8.0$ sec.

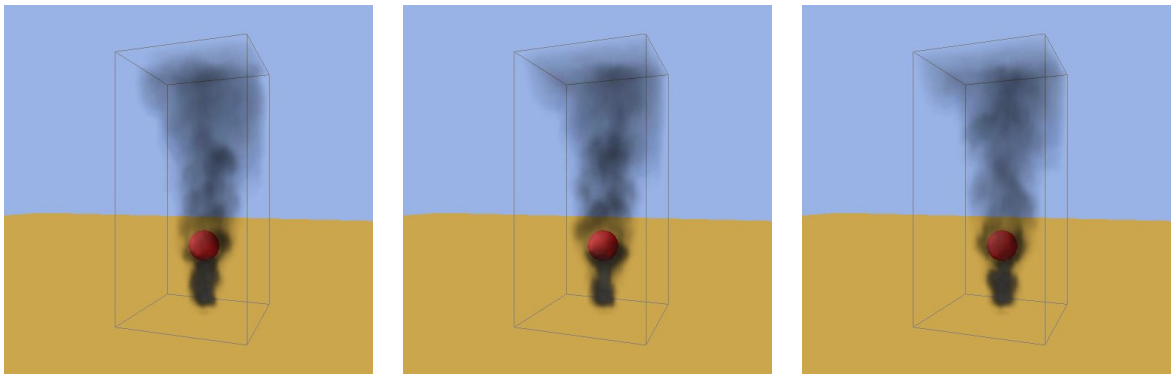
Figure 4: Example 1 – Rising smoke. The smoke has natural looking swirls due to the vorticity confinement, and the thermal buoyancy adds realism to the general smoke movement.



$t = 0.0$ sec. $t = 1.0$ sec. $t = 2.0$ sec.

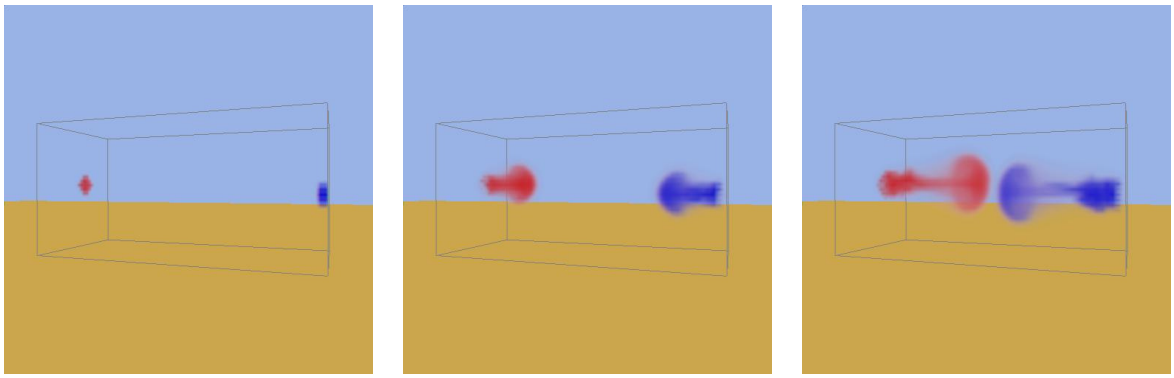


$t = 3.0$ sec. $t = 4.0$ sec. $t = 5.0$ sec.

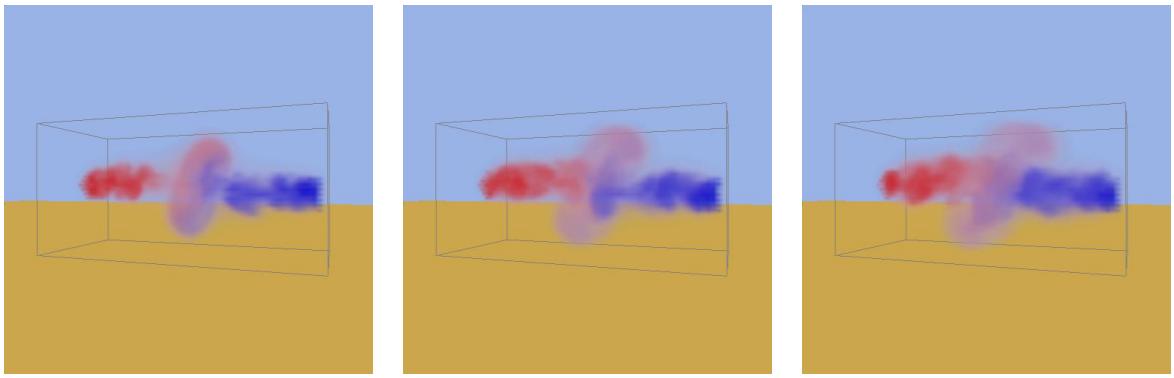


$t = 6.0$ sec. $t = 7.0$ sec. $t = 8.0$ sec.

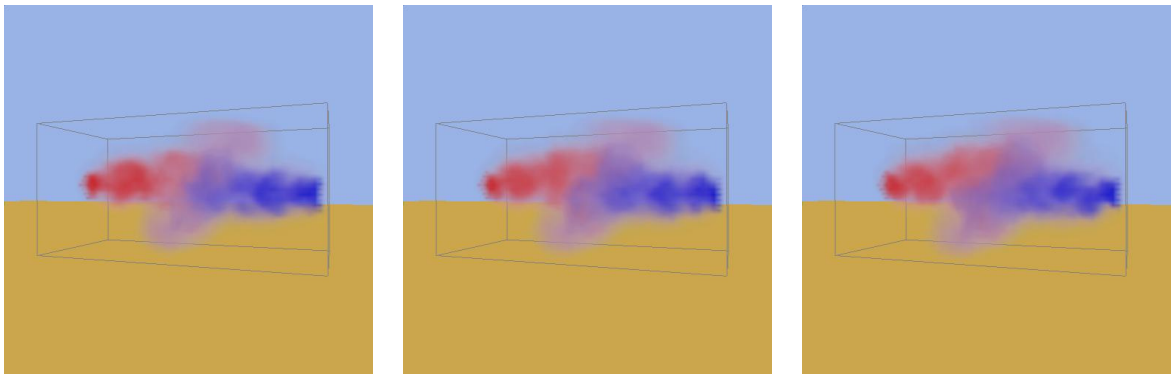
Figure 5: Example 2 – Interaction with a simple object. The smoke evolves a bit unnatural around the object due to the very simple handling of the boundary. However, an effect of interaction is achieved. The voxelized object is presented visually, using a spherical mesh, to improve visual quality.



$t = 0.0$ sec. $t = 1.0$ sec. $t = 2.0$ sec.



$t = 3.0$ sec. $t = 4.0$ sec. $t = 5.0$ sec.



$t = 6.0$ sec. $t = 7.0$ sec. $t = 8.0$ sec.

Figure 6: Example 3 – Multi-colored smoke. Two smoke streams with different colors meet and blend into a mixed color.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

Modelling and Simulation Tools

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

MODELLING OF FLUID PROPRIETIES IN HYDRAULIC POSITIVE DISPLACEMENT MACHINES

Paolo Casoli, Andrea Vacca, Germano Franzoni, Gian Luigi Berta

Dipartimento di Ingegneria Industriale

Università degli Studi di Parma
Parco Area delle Scienze 181/A
43100 Parma – Italy

Abstract

This paper presents a numerical model for the simulation of a swash-plate axial piston pump, focusing on the characterization of fluid proprieties.
As it is well known, the reduction of flow oddness (which generates pressure ripples and produces vibration and noise in the entire circuit) and the avoidance of cavitation are the major problems in the design of these pumps. Developing a simulation code can be very useful for component optimisation in order to predict and reduce the undesired phenomena. The paper first gives a quick overview on a previously developed pump model; afterwards four different models of the fluid are presented: they take into account cavitation in different ways. Their aim at characterizing as well as possible the unsteady and erratic cavitation features in a simplified manner, in order to apply the models to the simulation of hydraulic components.
In the second part of the paper these models are implemented in the simulation code of a particular piston pump: few results are presented and compared with available test data. The effects of the fluid models on the predict pump performance are shown and commented.

Keywords: pump, cavitation, fluid model

Nomenclature

B	Bulk Modulus
$\mathscr{L}_a$	Friction work
N	Number of cylinders
R	Bubble radius
R_G	Specific gas constant
$\tilde{R}_G$	Universal gas constant
T	Absolute temperature
V	Volume
$\dot{V}$	Volume flow rate
c	Velocity
m	Mass
$\dot{m}$	Mass flow rate
$\tilde{m}$	Oil vapour molecular mass
n	Shaft speed
p	Absolute pressure
t	Time
v	Specific volume
x	Gas to liquid volume fraction
y	Gas to liquid mass fraction
Ω	Section
α	Air to liquid volume fraction
ϕ	Vapour to liquid mass fraction
γ	Isentropic exponent
θ	Free gas to total gas fraction
ϑ	Angular position
ρ	Density
σ	Surface tension
ξ	Abscissa
$\dot{\xi}$	Linear velocity

Subscripts

A	Casing suction side
D	Delivery
G	Gas
L	Liquid
M	Casing delivery side
S	Suction
SAT	Saturation conditions
V	Vapour
VAP	Saturated vapour
$VAPH$	Higher saturated vapour
$VAPL$	Lower saturated vapour
c	Cylinder
i	Index
id	Ideal
s	Static condition
t	Total condition
0	Reference condition

Abbreviations

IDC	Inner dead center
ODC	Outer dead center

Introduction

Axial piston pumps are widely used in modern hydraulic circuits, thanks to their high specific power, efficiency and reliability [1]. Despite these advantages, the pump operation is affected by a few drawbacks. In fact, in positive displacement machines, both the pumping mechanism and the fluid compressibility induce a flow oddness that determines pressure pulsations and leads to vibrations and noise through out the circuit [2, 3].
Besides, when the pump operates taking oil from a tank, the fluid could be in contact with atmospheric air that partially dissolves in the liquid. During the suction phase the fluid pressure reduction due to pressure losses and, in particular, dynamic effects can cause the air to be released in bubbles [4, 5]. If the pressure further decreases, besides air-release, vapour cavitation can occur. Cavitation causes the volumetric efficiency to drop and can damage some parts.

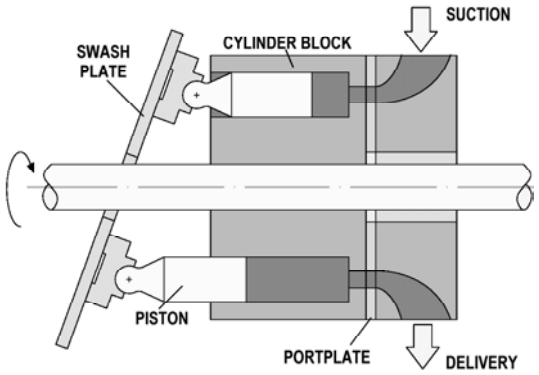


Figure 1 – Schematic of swash plate piston pump

The main targets concerning research and development of these machines are the reduction of the flow ripple and the avoidance of cavitation.
This goal can only be attained by the use of computer-aided optimization of present designs. In this context the authors developed a numerical model for the simulation of axial piston pumps. In order to reach a detailed description of the pump operating condition, the model focuses on several important aspects; in particular on the modelling of the pump geometry and fluid proprieties.
The model was implemented in a numerical code in FORTRAN language and was validated on the basis of some experimental data [5, 6].

The Pump Model

Figure 1 shows the basic structure of a swash plate axial piston pump. The numerical model is based on a finite volume concept, according to the *Filling & Emptying* approach: the pump is divided in number of chambers where the fluid is assigned uniform proprieties. Figure 2 represents the framework adopted for a single cylinder pump. Two constant volumes, V_A and V_M, represent the internal cavities in the pump casing at the suction and delivery side respectively, while the cylinders volume are variable as a result of the pistons motion.

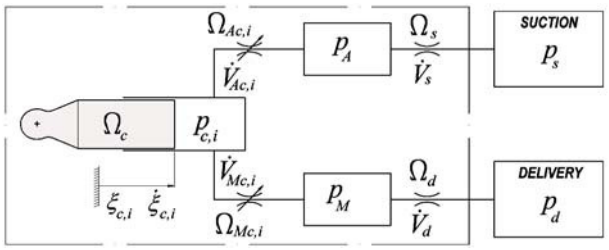


Figure 2 – Basic model for a single cylinder pump

Three modelling features have been carefully considered and deepened, namely portplate geometry, unsteady fluid dynamics and physical fluid properties [6-8].
When the delivery port suddenly opens, the pressure ripple is produced by the sharp contact of high pressure in $V_{M,I}$ with low pressure in $V_{c,i}$. The pressure peak can be made smoother through a smart design of the port plate. Hence the port plate is carefully modelled.

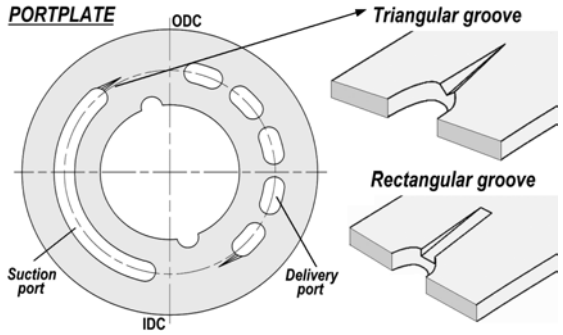


Figure 3 – Portplate and groove different geometries

The throat areas of the variable orifices $\Omega_{Ac,i}$ and $\Omega_{Mc,i}$ depend on the position of the cylinder block with respect to the portplate. This latter (fig. 3) presents two kidney ports connected to the suction and delivery volumes and two silencing grooves that allow the cylinder to pass smoothly from inlet to delivery. The groove design has a relevant effect on flow and pressure fluctuations [6, 7, 9, 10]: their dimensions, shape, slope and length are parameters that have to be optimised to reduce this

pressure peaks. For this purpose an accurate technique, named as "*lined surface method*" [6-8], has been implemented to calculate the instantaneous throat area, generated by the superimposition of either the groove or the kidney port to the cylinder block port. The model allows the user to define a generically shaped groove, e.g. fig. 3 reports the case of a triangular and a rectangular groove with straight axis.

In the fluid dynamic model the mass exchange occurs between the permeable parts of the control surfaces of each chamber (that define the area of the orifices in fig. 2). The fluid flow is calculated by the generalized Bernoulli's equation, under quasi-steady state and isothermal conditions:

$$c\mathrm{d}c = -v\mathrm{d}p - \mathrm{d}\mathscr{L}_a \qquad (1)$$

The equation is solved on the basis of the function *v(p)* that depends on the adopted fluid model, while including $\mathrm{d}\mathscr{L}_a$ in the discharge coefficients.

The next paragraph describes different solutions corresponding to four different fluid models.

Considering continuity equation, the pressure course inside each chamber is determined; in particular, for the cylinder volume, one obtains:

$$\sum \dot{m}_{c,i} + \frac{V_{c,i}}{\left(\frac{\partial p}{\partial \rho}\right)_T}\frac{\mathrm{d}p_{c,i}}{\mathrm{d}t} - \rho_{c,i}\Omega_c\dot{\xi}_{c,i} = 0 \qquad (2)$$

The first term is related to the mass exchange with the adjoining volumes, the second term is related to compressibility effects, under isothermal conditions, and the last term considers the rate of volume variation produced by the piston motion.

The model has been developed also considering the rate of change of momentum of the fluid in the region of the delivery groove [9, 10]. When a cylinder is switching from suction to delivery phase, the edge of the cylinder block port starts uncovering the groove: the dynamic effects, due to the sudden acceleration of fluid in the restriction, induce further pressure pulsations inside the cylinder. On this process, the model calculates the flow rate as an independent state variable from the momentum equation, under the hypothesis of 1-D uncompressible flow [7]. Consequently the cylinder pump is described with 4 differential equations, when the cylinder port is uncovering the delivery groove the flow rate is calculated as a state variable from the momentum equation, otherwise it depends on the ratio between upstream and downstream pressure. The code allows the user to define a pump with a generic number *N* of pistons, so that the system is described with 2*N*+2 state equations when the groove is working, and *N+2* equations otherwise, as reported in [7]. The code integrates all the equations by means of a fourth-order Runge-Kutta method with step size control [11].

Modelling fluid proprieties

The authors developed four different models (in the paper simply identified with the letters **A**,**B**,**C**,**D**) of the fluid with the main target of simulating in a simplified manner the behaviour of fluid when both gaseous and vapours cavitation occur. These models are based on continuous mathematical functions with continuous first derivatives, in order to avoid numerical instability.

Model A is the simplest, the fluid is treated as a uniform gas-liquid mixture, vapour cavitation is not considered.

The bulk modulus of the liquid is constant; surface tension effects are neglected; fluid pressure and temperature are the same for both gas and liquid.

The assumption of isothermal fluid is consistent with no phase change in the oil, and supported by the higher heat capacity of liquid comparatively with gas. The gaseous phase is modelled by the state equation of ideal gases, thus neglecting both the contribution of oil vapour and the effect of high pressure on compressibility factor.

The mixture volume, for mass unit, is given by:

$$v = v_L\left(1-y\right) + v_G \cdot y \qquad (3)$$

Where, the gas to liquid mass fraction is:

$$y = \frac{\rho_{G0}V_{G0}}{\rho_{G0}V_{G0} + \rho_{L0}V_{L0}} \cong \frac{v_L}{v_G}x \qquad (4)$$

The state equation of the mixture becomes:

$$v = \frac{R_G \cdot T_G}{p}\cdot y + \frac{1-y}{\rho_{L0}\left(1+\frac{p-p_0}{B}\right)} \qquad (5)$$

The expression of the first derivative of the pressure with respect to the density is:

$$\left(\frac{\partial p}{\partial \rho}\right)_T = \frac{\left[\frac{R_G T_G}{p}\cdot y + \frac{1-y}{\rho_{L0}}\left(1-\frac{p-p_0}{B}\right)^{-1}\right]^2}{\frac{R_G T_G y}{p^2} + B\left(1-y\right)\left[\rho_{L0}\left(1+\frac{p-p_0}{B}\right)\right]^{-2}} \qquad (6)$$

Substituting eq. (6) into eq. (1) and integrating, the ideal fluid velocity at the throat results:

$$c_{id} = \int_{p_s}^{p_t} v\,dp = R_G T_G y \cdot \ln\left(\frac{p_t}{p_s}\right) + \frac{B(1-y)}{\rho_{L0}} \cdot \ln\left(\frac{\rho_{L0}\left(1+\frac{p_t - p_0}{B}\right)}{\rho_{L0}\left(1+\frac{p_s - p_0}{B}\right)}\right) \quad (7)$$

Model B assumes that the air can partially dissolve into the oil, according to Dalton-Henry's law. The non-dissolved gas volume fraction depends on the fluid pressure and it is uniformly mixed with the liquid. Vapour cavitation is not considered.

In the case that as far as $\boldsymbol{p > p_{SAT}}$ the gas is completely dissolved in the liquid ($\theta = 0$), so that the Bulk modulus is not affected by any gas. Since the gas gives only a contribution of mass (and not of volume), the fluid density (at reference conditions) is [2, 12]:

$$\rho_0 = \rho_{L0} + \frac{x}{1-x}\rho_{G0} \quad (8)$$

If both temperature and pressure vary, the mass remains constant but the volume changes; considering the liquid bulk modulus B the following expression is obtained:

$$\rho(p)_{T=T_0} = \rho_0 \cdot e^{\frac{p-p_0}{B}} \quad (9)$$

When $\boldsymbol{p < p_{SAT}}$ only a part of the gas is dissolved. The remaining free gas fraction follows the Henry's law:

$$\theta = \frac{p_{SAT} - p}{p_{SAT}} \quad (10)$$

Consequently the $\theta(p)$ function's first derivative is not continuous:

$$\frac{d\theta}{dp} = \begin{cases} 0 & \text{if } p \geq p_{SAT} \\ -1/p_{SAT} & \text{if } p < p_{SAT} \end{cases} \quad (11)$$

To handle the discontinuity and avoid the numerical problems, the function $\theta(p)$ has been substituted by a polynomial expression whose first derivative is forced to zero at $p = 0$ bar, and at $p = p_{SAT}$ [12].

Starting from eq. (10), the fluid density was found as a function of pressure and temperature, under the assumptions of liquid density independent of temperature and of constant bulk modulus:

$$\rho(p,T) = \frac{(1-x)\rho_{L0} + x\rho_{G0}}{(1-x)e^{\left(\frac{p_0-p}{B}\right)} + \theta\frac{xT}{T_0}\left(\frac{p_0}{p}\right)^{1/\gamma}} \quad (12)$$

The free gas was considered following an isentropic process (identified by exponent γ) from the reference (p_0, T_0) to actual (p, T) conditions.

The first derivative is given by:

$$\left(\frac{\partial p}{\partial \rho}\right)_T = \begin{cases} \frac{B}{\rho_0} e^{\frac{p_0-p}{B}} & p \geq p_{SAT} \\ -\frac{\Psi}{\Pi}\left[-\frac{1-x}{B}e^{\frac{p_0-p}{B}} + \frac{Tp_0^{1/\gamma}}{T_0}\cdot \left(\frac{d\theta}{dp}xp^{-1/\gamma} - \frac{x\theta}{\gamma}p^{-1/\gamma-1}\right)\right]^{-1} & p < p_{SAT} \end{cases} \quad (13)$$

where:

$$\Pi = (1-x)\rho_{L0} + x\rho_{G0}$$

$$\Psi = \left((1-x)e^{\frac{p_0-p}{B}} + \theta\frac{xT}{T_0}\left(\frac{p_0}{p}\right)^{\frac{1}{\gamma}}\right)^2 \quad (14)$$

Eq. (13) points out that the first derivative is well defined for $p = p_{SAT}$ because the left and right limits exist and are equal.

For $p > p_{SAT}$ the integration of eq. (1) yields:

$$c_{id} = \int_{p_S}^{p_T} v\,dp = \rho_0 B\left[e^{\frac{p_t-p_0}{B}} - e^{\frac{p_s-p_0}{B}}\right] \quad (15)$$

If $p < p_{SAT}$ the solution is found by means of a numerical algorithm.

Model C upgrades the previous one considering also vapour cavitation. Since the oil is a mixture of components of different chemical nature, the phase change occurs in a pressure range defined by a higher (p_{VAPH}) and a lower (p_{VAPL}) limit [12]. In other words the oil evaporation starts just below the upper limit and completely finishes when the lower pressure value is reached. The values assumed for p_{VAPL}, p_{VAPH}, have been deduced from [12-14].

As far as $\boldsymbol{p > p_{SAT}}$ the fluid is described by means of the same equations of model B, and the parameter θ is null.

If $\boldsymbol{p_{VAPH} \leq p \leq p_{SAT}}$ the model C still agrees with model B, but it is assumed that the gas is completely free when the higher vapour pressure is reached. In fact:

$$\theta = \frac{p_{SAT} - p}{p_{SAT} - p_{VAPH}} \tag{16}$$

Eq. (16) shows that $\theta = 0$ if $p = p_{SAT}$ while $\theta = 1$ if $p \le p_{VAPH}$.

When $\boldsymbol{p_{VAPL} < p < p_{VAPH}}$ the fluid is treated as a uniform mixture of free air, oil vapours and liquid oil. In this pressure range the vapour to liquid mass fraction was describe with a polynomial function:

$$\phi = \frac{m_V}{m_L} = \begin{cases} 0 & p \ge p_{VAPH} \\ \phi(p) & p_{VAPH} > p > p_{VAPL} \\ 1 & p \le p_{VAPL} \end{cases} \tag{17}$$

Where $\phi(p)$ was chosen according to the data reported in [12-14]. On the basis of eq. (17), being $\theta = 1$ in the pressure range considered, the fluid density can be found as a function of pressure and temperature, under the hypotheses of liquid density independent of temperature and constant bulk modulus. It was also assumed that both the free gas and the vapour phase undergo an isentropic process [12]:

$$\rho = \frac{(1-x)\rho_{L0} + x\rho_{G0}}{\left\{ \begin{array}{l} (1-x)(1-\phi)e^{\frac{p_0 - p}{B}} + \frac{T}{T_0} \cdot \\ \cdot \left[\frac{\rho_{L0}(1-x)\phi}{\rho_{V0}} \left(\frac{p_{VAPH}}{p} \right)^{\frac{1}{\gamma}} + x \left(\frac{p_0}{p} \right)^{\frac{1}{\gamma}} \right] \end{array} \right\}} \tag{18}$$

where: $$\rho_{V0} = \frac{\tilde{m}}{\tilde{R}_G \cdot T_0} p_{VAPH} \tag{19}$$

When the fluid pressure decreases below the lower saturation vapour pressure ***($p \le p_{VAPL}$)***, only the gaseous phase (mixture of non-dissolved air and oil vapour) exists. The density can be easily determined setting $\phi = 1$ in eq. (18).

If $p > p_{SAT}$ the expression of $\partial p / \partial \rho$ is the same as in eq. (13) considering the value of θ given by eq. (16). In the other cases:

if $p_{VAPL} \le p \le p_{VAPH}$:

$$\left(\frac{\partial p}{\partial \rho} \right)_T = -\frac{\Psi}{\Pi} \left[\begin{array}{l} (1-x)\left(\frac{1-\phi}{B} - \frac{d\phi}{dp} \right) e^{\frac{p_0 - p}{B}} + \\ + \frac{\partial \phi}{\partial p} C_1 p^{-\frac{1}{\gamma}} - \frac{C_1 + \phi C_2}{\gamma} p^{-1-\frac{1}{\gamma}} \end{array} \right]^{-1} \tag{20}$$

if $p < p_{VAPL}$:

$$\frac{\partial p}{\partial \rho} = \frac{1}{\gamma} \frac{\Pi}{(C_1 + C_2) p^{\frac{\gamma - 1}{\gamma}}} \tag{21}$$

where $C_1 = x \frac{T}{T_0} p_0^{\frac{1}{\gamma}}$ and $C_2 = (1-x) \frac{T}{T_0} p_{VAPH}^{\frac{1}{\gamma}}$. In this latter case from eq. (1) one obtains:

$$\begin{aligned} c_{id} &= \int_{P_s}^{P_t} v \mathrm{d}p = \\ &= \frac{C_1 + C_2}{\Pi} \frac{\gamma}{\gamma - 1} \left[(p_t)^{\frac{\gamma - 1}{\gamma}} - (p_s)^{\frac{\gamma - 1}{\gamma}} \right] \end{aligned} \tag{22}$$

Model D considers the liquid and gaseous phase as separated: the gas is contained in the liquid in the form of spherical bubbles with an equal radius *R*. The pressure equilibrium between the inside of the bubble and the external environment, at the reference conditions is given by [4,14]:

$$p_{G0} = p_{L0} - p_{VAP} + \frac{2\sigma}{R_0} \tag{23}$$

The air to liquid volume fraction is $\alpha = V_G / V_L$, by virtue of the perfect gas state equation it can be expressed as:

$$\frac{\alpha}{\alpha_0} = \frac{p_{L0} - p_{VAP} + 2\sigma / R_0}{p_L - p_{VAP} + 2\sigma / R} \tag{24}$$

Neglecting the temperature variation, α can be expressed as a function of the liquid pressure:

$$\alpha = \alpha_0 \frac{p_{L0} - p_{VAP} + \frac{2\sigma}{R_0}}{p_L - p_{VAP} + \frac{2\sigma}{R_0 (\alpha / \alpha_0)^{1/3}}} \tag{25}$$

Obviously eq. (25) is not significant if the liquid pressure is lower than the saturation pressure because only the vapour phase exists [4].

Eq. (25) is not explicit, so that the solution must be reached by a numerical procedure [11].

The fluid density is determined starting from the mass to volume fraction, and neglecting the contribution of gas mass with respect to liquid:

$$\rho = \frac{m_G + m_L}{V_G + V_L} \approx \rho_L \frac{1}{\alpha + 1} = \rho_{L0} e^{\frac{p_L - p_0}{B}} \frac{1}{\alpha + 1} \tag{26}$$

The function $\partial p / \partial \rho$ can expressed as follows:

$$\left(\frac{\partial p}{\partial \rho} \right)_T = \left(\frac{\partial p}{\partial \alpha} \frac{\partial \alpha}{\partial \rho} \right)_T \tag{27}$$

where $\left(\frac{\partial \rho}{\partial \alpha} \right)_T = \frac{\partial \rho_L}{\partial \alpha} \left(\frac{1}{\alpha + 1} \right) - \frac{\rho_L}{(\alpha + 1)^2}$,

$$\left(\frac{\partial p}{\partial \alpha}\right)_T = \frac{2\sigma\alpha_0^{1/3}}{R_0}\alpha^{-4/3} - \frac{\alpha_0}{\alpha^2}\left(p_L - p_{VAP} + \frac{2\sigma}{R_0}\right)$$

Model D is defined only over the p_{VAP} value. This model presents a few drawbacks due to the difficult choice of values for both the bubble radius and the surface tension.

Fluid model comparison

In order to present a significant comparison many common parameters were set at the same value (see tab.1).

	model			
	A	**B**	**C**	**D**
T [K]	311	311	311	311
p_{SAT} [bar]	-	1	1	1
p_{VAPH} (p_{VAP}) [bar]	-	-	0.3	0.3
p_{VAPL} [bar]	-	-	0.25	-
x [%]	9	9	9	9
ρ_{L0} [kg/m³]	850	850	850	850
ρ_{G0} [kg/m³]	1.2	1.2	1.2	1.2
$\tilde{m}$ [kg/mol]	-	-	200	-
R_0 [m]	-	-	-	0.0005
σ [Pa]	-	-	-	0.03

Table 1 – Values of main parameters for model comparison

In the case of model D, $p_0 = p_{SAT}$ and $\alpha_0 = x$; the saturated vapour pressure was set to an intermediate value between p_{VAPH} and p_{VAPL} adopted for the model C.

Fig. 4 shows fluid density versus pressure: models B and C show a similar trend and the cavitation is predicted to appear at a lower pressure comparatively to model A. In fact eq. (10) confirms how the air fraction increases while the pressure decreases; on the contrary in model A the air fraction is always constant. With model D cavitation occurs at the highest pressure: this strongly depends on the assumed value for the vapour saturation pressure. Fig. 5 shows $\partial p/\partial \rho$ as a function of pressure: model C has not a constantly increasing trend as the others. In the range between p_{VAPL} and p_{SAT} the function presents a particular course with two minimum points: this is due to the presence of a two-phase fluid (liquid and gas) where the gas phase is a mixture of air and oil vapour. The figure also shows a different trend for the models B and C, while fig. 4 indicates that the function $\rho(p)$ is similar for the two cases.

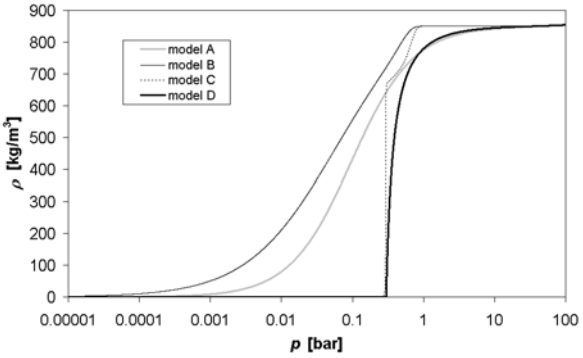


Figure 4 – Simulated fluid density

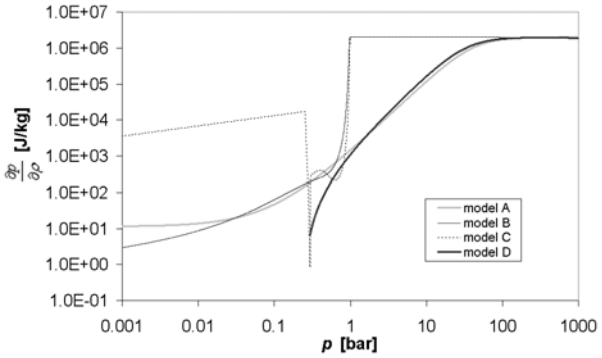


Figure 5 – Simulated $\partial p/\partial \rho$ courses

Number of cylinders	9
Port plate timing angle	5°
Number of inlet slots (Fig. 3)	1
Number of delivery slots (Fig. 3)	5
Displacement	45.47 cm³
Max speed	2600 r/min
Max delivery continuous pressure	280 bar
Mass (without oil)	24 kg

Table 2 – Main features of the pump Casappa LVP48

Results

The previously described pump simulation code can be linked with each of the four fluid models. Four FORTRAN libraries were created for calculation of fluid physical properties (i.e. density, sound speed, etc.). A comparison with several set of experimental data has been very useful to find out which fluid model is more suitable for the simulation of axial piston pumps. Each model requires a few calibration parameters; the results presented in the following refer to the set of parameters displayed in tab. 1. Experiments were carried out on a stock pump (Casappa LVP48, see tab. 2), during several tests campaign. Simulation results have been compared both with pressure measured inside the delivery volume of the pump (reported in [15], and the pump speed characteristic $\left(n, \dot{V}\right)$ (provided by Casappa S.p.A.).

Among calibration parameters the coefficients of flow discharge were chosen from the data reported in [16] for similar conditions. Later they were slightly adjusted (where the actual throat area differs significantly from case reported in [16]) to better match experimental data, referring to the fluid model C. Results obtained with model D are not presented, because of his strong sensitivity to fluid parameters (e.g. bubble radius), that makes the model impractical heretofore.

Fig. 6 shows a comparison between experimental data and simulation results obtained with models A and C, highlighting a good agreement of the latter. By contrast carrying out the simulation with model A, a strong pressure overshoot occurs. Although coefficients of discharge were calibrated with model C, discrepancies with model A do not appear to depend of this procedure, so far as a reasonable set of coefficient was not possible to find.

Model A considers the air always un-dissolved, while, for $p > p_{SAT}$, model C considers the air completely dissolved in the liquid; this presence of free air induces stronger oscillations.

Experimental conditions have been sometimes changed, in order to verify the reliability of the adopted set of calibration parameters. E.g. fig. 7 reports the comparison at a different shaft speed.

Fig. 8 shows speed characteristic of the pump: also here a good agreement with experimental data is obtained using model C. At higher shaft speeds, model C shows a significant reduction of the flow rate gradient: during the suction phase, the pressure inside the cylinder reaches a lower value than the saturation pressure. This causes an increase of the free air fraction and consequently a reduction in fluid density, as shown in fig 4.

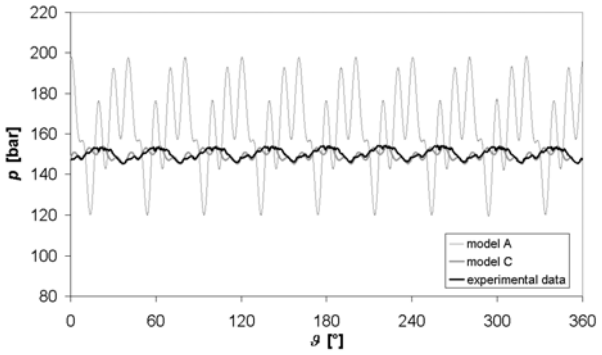


Figure 6 – Delivery port pressure (n = 2000 r/min, $p_D - p_S$ = 150 bar; T = 38°C)

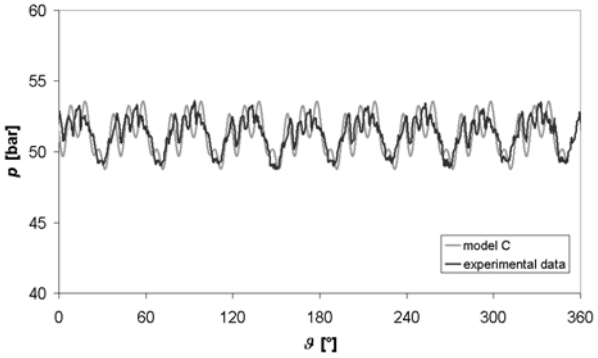


Figure 7 - Delivery port pressure (n = 500 r/min, $p_D - p_S$ = 50 bar; T = 39°C)

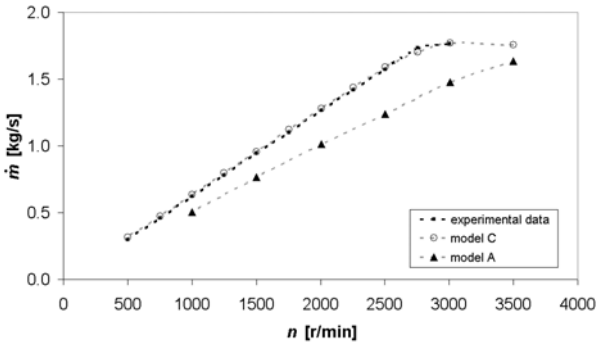


Figure 8 – Simulated and experimental pump characteristics

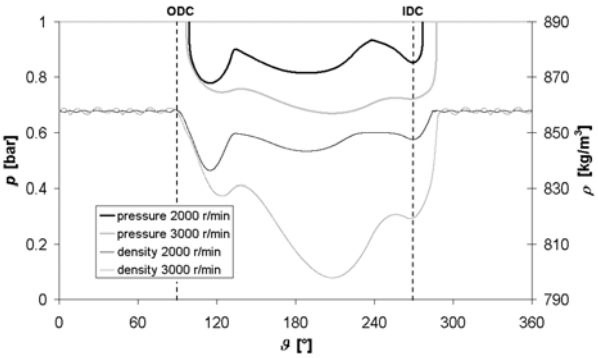


Figure 9 – Cylinder pressure and density during suction phase

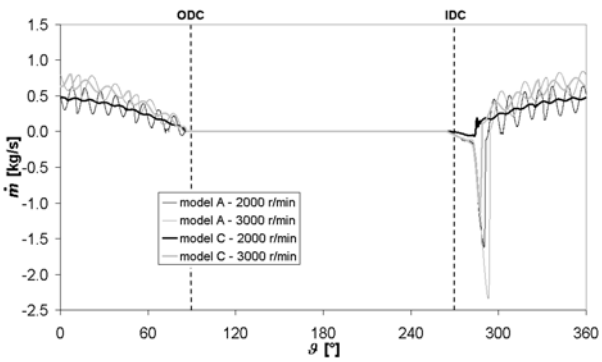


Figure 10 – Mass flow rate outgoing from the cylinder during the delivery phase

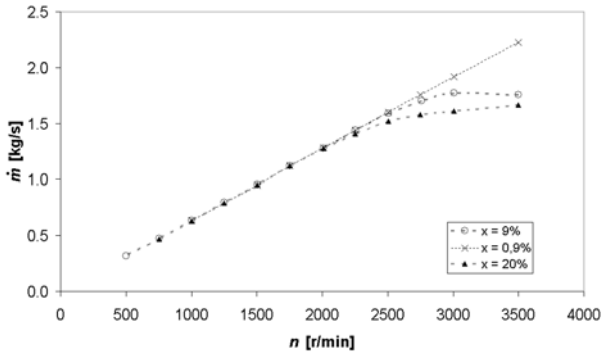


Figure 11 – Pump characteristics for different values of parameter x

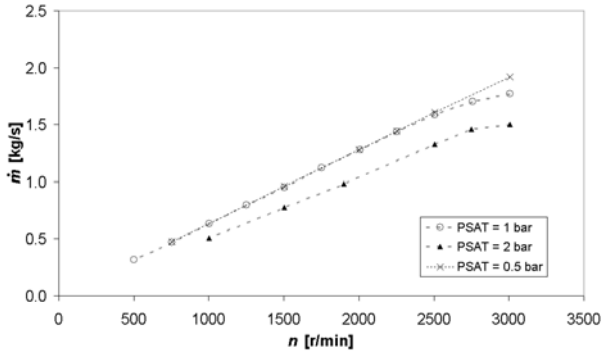


Figure 12 – Pump characteristics for different values of parameter p_{SAT}

Using model A the presence of undissolved air yields lower mass flow rates throughout the range considered, because the fluid density is always lower. Using model A the numerical curve does not describe the flow rate gradient revealed by experiments at about 2800 r/min, due to the unset of cavitation.

If the shaft speed increases, during the suction stroke the piston motion causes an increase in the cylinder volume that could not be compensated by the incoming volume of fluid; as a result pressure decreases and cavitation may occur.

Since experiments were carried out with p_S = 1 bar, saturation pressure was assumed equal to inlet pressure; consequently gaseous cavitation always occurs because $p_{c,i} < p_{\mathrm{SAT}}$ during the suction stroke; this phenomenon is emphasized at higher speed.

The effects of the shaft speed are shown, referring to model C in fig. 9, where a density reduction, due to a pressure reduction inside the cylinder, during suction, is clearly visible.

Fig 9 shows that, during a revolution, the cylinder pressure decreases below the assumed saturation pressure but remains above the higher saturated vapour pressure (model C), so that only gaseous cavitation is predicted to occur. It was observed that vapour cavitation appears only at very high speed, outside the practical range of application of the pump. Consequently the same results obtained with the complex model C can be reached also adopting model B that is simpler and makes the simulation faster.

As said before, vapour cavitation could occur if the cylinder pressure reached the p_{VAPH} value, during the suction phase. In these conditions model C would highlight a sudden decrease of density (fig. 4); hence the delivery mass flow rate would be lower than the same one calculated with model A, differently from the cases taken into account in this paper.

Therefore, while model C agrees with experimental data, model A overestimates cavitation effects in the typical operating conditions of the pump (fig. 8) and undervalues these effects at the highest speeds.

The presence of gaseous cavitation makes the pump performance worse: the lower pressure reached inside the cylinder causes a significant backflow (as displayed in fig. 10, referred to different rotational speeds), influencing the flow ripple intensity. This phenomenon looks more relevant with model A (that predict a lower cylinder pressure).

Fig. 10 shows the delivery mass flow rate for a cylinder: when the piston reaches the IDC, the cylinder slot starts discovering the silencing groove placed at the beginning of the portplate delivery port (fig. 3). The initial flow rate is negative because the cylinder pressure is lower than the delivery pressure and the fluid enters into the cylinder volume.

The amount of predicted backflow depends on several factors: such as the portplate geometry, the pump speed and the adopted fluid model. Fig. 10 shows the fluid model and speed influence: adopting model A gaseous cavitation effect leads high backflow in both considered speed, on the contrary with model C such a negative flow rate occurs only at highest speed.

In fig. 11 the effects of the parameter x on the characteristics curve is presented using the fluid model C. At the lowest value of x (x = 0.9%) the presence of air is negligible and the curve is linear. Increasing the air content the effects are clearly visible: the gradient reduction begins at lower shaft speed, and the showed flow rate is lower.

In fig. 12 the effect of changes in the saturation pressure, using model C, is presented. Reducing p_{SAT} the gaseous cavitation starts at lower pressures. If p_{SAT} = 0.5 bar, gaseous cavitation does not occur and the fluid behaviour is similar to a pure liquid because the lowest pressure reached inside the cylinder is about 0.6 bar (fig. 9). When p_{SAT} is set to 1 bar the gaseous cavitation is predicted to appear and the flow rate gradient is reduced over 2500 r/min. At the highest value of the saturation pressure (p_{SAT} = 2 bar) gaseous cavitation is always presents and flow rate is affected at all speeds.

Conclusion

Four hydraulic fluid models (simply named as A,B,C and D) have been presented for the simulation of hydraulic components. The models differ for the manner they take into account cavitation phenomena: not all consider the vapour cavitation, while the gaseous cavitation is treated with different approaches. The authors developed a numerical code for the simulation of swash-plate axial piston pumps, in which case simulation can be very useful for the optimisation of component geometry, so as to predict and reduce the flow ripple and avoid cavitation.

In order to evaluate which model is more suitable for pump analysis, several simulations were carried out using more than one model and comparing the results with some experimental data available. It was observed that model C, that takes into account both gaseous and vapour cavitation, agrees with experiments on a wider range of conditions. Nevertheless, considering the usual operating conditions of the analyzed pump it was observed that vapour cavitation is never likely to occur; therefore model B can be used as well, thus reducing the simulation time.

The results presented in the paper also show that model A overestimates cavitation so far as only gaseous cavitation occurs, while undervalues the phenomena when both gaseous and vapour cavitation take place.

References

[1] Nervegna N. *Oleodinamica e Pneumatica*, Torino, Italia: Politeko, Marzo 2000.

[2] E. Benjamin Wylie, Victor L. Streeter *Fluid Transient in Systems* Prentice Hall, Englewood Cliffs,1993.

[3] K.A. Edge, J. Darling, *A theoretical model of axial piston pumps flow ripple*, In: Proceeding of First Bath International Fluid Power Workshop, Bath, U.K., 1988, pag. 113 ÷136.

[4] Tillner W., Fritsch H., Kruft R., Lehman W., Masendorf D. G. *The avoidance of cavitation damage***,** London: MEP, 1993.

[5] Isay W. H., *Kavitation* Hamburg: Schiffahrts-Verlag 1981.

[6] G.L. Berta, P. Casoli, A. Vacca, M. Guidetti *Simulation model of axial piston pumps inclusive of cavitation.* ASME International Mechanical Engineering Congress and Exposition, New Orleans, Lousiana, USA, November 17-22 2002.

[7] Casoli P., Vacca A. *Modeling of an axial piston pump for pressure ripple analysis.* In: Proceeding of The Eight Scandinavian International Conference on Fluid Power, Tampere Finland May 7-9, 2003.

[8] Casoli P., Vacca A. *Modello di Simulazione di una Pompa a Pistoni Assiali*, Atti del Dipartimento di Ingegneria Industriale, Pub- 01-1, 2001.

[9] A.M. Harrison, K.A. Edge, *Reduction of axial piston pump pressure ripple*, In: Proc Instn. Mech. Engrs, 2000, Vol 214 Pt I.

[10] K.A. Edge, J. Darling, *The pumping dynamics of swash plate piston pumps*, In: Journal Of Dynamic Systems, Measurement and Control, Trans. of ASME, June 1989, Vol. 111, pag. 307÷312.

[11] J.D. Hoffman *Numerical methods for engineers and scientists,* Mc Graw Hill, 1992.

[12] IMAGINE s.A. *AMESim® Standard fluid properties, – Indexed elementary fluid properties* Report FP01. Roanne, France, 1995-2000.

[13] Wallis G. B. – *One dimensional two-phase flow* – Mc Graw Hill, 1969.

[14] Plesset M. S., *Tensile strength of liquids*, California institute of technology, Rep 85-47, 1969.

[15] Mancò S. Nervegna N. Lettini A. Gilardino L. *Advances in the simulation of axial piston pumps* In: Proceeding of the 5th JFPS International Symposium on Fluid Power, Nara, Japan, 12-15 November, 2002.

[16] E. Idel'cik *Memento del Pertes de Charge, Coefficients de Pertes de Charge Singulières et de Pertes de Charge par Frottement*, Edition Eyrolles,1986.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

GENERALIZED NET MODEL OF STOREHOUSE PROCESSES: A PRINCIPAL SCHEME

Nikolinka G. Christova*
University of Chemical Technology and Metallurgy
Department of Automation of Industry
8 Blvd. Kliment Ohridski, 1756 Sofia
Bulgaria

Krassimir T. Atanassov
Bulgarian Academy of Sciences
CLBME, Acad. G. Bonchev Str., bl. 105, 1113 Sofia
Bulgaria

Abstract

The Generalized Nets (GNs) are extensions of Petri nets. In the last ten years they were applied in many theoretical and application areas. Here, for a first time they will be used for modelling of storehouse processes. A new approach of modelling storehouse processes by implementation of generalized nets will be proposed and investigated. Detailed description of the developed GN-model will be presented. The ability and the interpretability of the GN-based model for effective management of storehouse processes will be also discussed.
Keywords: Modelling, Generalized Nets, Storehouse processes

Introduction

The nature of discrete-event simulation [1], with its modelling of systems of interacting queues, is clearly of wide applicability.

The storehouses are one of the basic components of every company. In every storehouse different commodities are accepted, preserved and dispatched; plans for the storehouse functioning are prepared; requests for commodities are rendered or rejected; load/unload works are accomplished.

In the present paper for a first time the storehouse processes will be described by implementation of Generalized Nets (GNs), see [2]. The GNs are extensions of the ordinary Petri nets and the other Petri net modifications [3].

* Corresponding author. Phone: +359 888 602070, Fax: +359 2974 3867, E-mail: n_christova@automation.uctm.edu

Generalized Nets [2] are very useful model that provide a simple graphical representation of hybrid systems and takes advantage of the modular structure of the nets in giving a compact description of the system. They could be used as proper tools for modelling, simulation and forecasting of the processes, real-time control and/or optimization of non-critical in time real-time processes [2].

In Second section short remarks on generalized nets will be provided. In Third section we will describe a generalized net, which represents the functional model of storehouse activities. On the basis of this generalized net, the storehouse processes can be:

- simulated,
- controlled,

- optimized (this direction of the usage of generalized nets is not discussed here).

The model is very simple. In the present form, it is only an illustration of the possibilities of generalized nets to model a storehouse. If we would like to detailize the processes, we shall use other GN-possibilities, e.g., some of the hierarchical operators defined over generalized nets. This model will be an object of a future research of the authors.

Short remarks on generalized nets

The concept of a Generalized Net (GN) is described in [2]. Certain GNs may not contain the whole set of components; thus giving rise to special classes of GNs called "reduced GNs". For the needs of the present research we shall use (and describe below) one of the reduced types of GNs.
Formally, every transition of this reduced class of GNs is described (Figure 1) by $Z = \langle L', L'', r, \Box \rangle$, where:

(a) L' and L'' are finite, non-empty sets of places (the transition's input and output places, respectively). For the transition in Figure 1 these are $L' = \{l'_1,...,l'_m\}$ and $L'' = \{l''_1,...,l''_n\}$.

(b) r is the transition's condition determining which tokens will pass (or transfer) from the transition's inputs to its outputs; it has the form of an Index Matrix (IM; see [2]):

$$r = \begin{array}{c|ccc} & l''_1 & l''_j & l''_n \\ \hline l'_1 & & & \\ & & r_{i,j} & \\ l'_i & & (r_{i,j} = predicate, & \\ & & 1 \le i \le m, 1 \le j \le n) & \\ l'_m & & & \end{array}; \qquad (1)$$

where $r_{i,j}$ is the predicate which corresponds to the i-th input and j-th output places. When its truth-value is "true", a token from the i-th input place can be transferred to the j-th output place; otherwise, this is not possible;

(c) $\Box$ is a Boolean expression. It may contain as variables the symbols which serve as labels for transition's input places, and it is an expression consisting of variables and the Boolean connectives $\wedge$ and $\vee$ whose semantics is defined as follows:

$\wedge(l_{i_1},...,l_{i_u})$ - every place $l_{i_1},...,l_{i_u}$ must contain at least one token;

$\vee(l_{i_1},...,l_{i_u})$ - there must be at least one token in any of the places $l_{i_1},...,l_{i_u}$, where $\{l_{i_1},...,l_{i_u}\} \subset L'$.

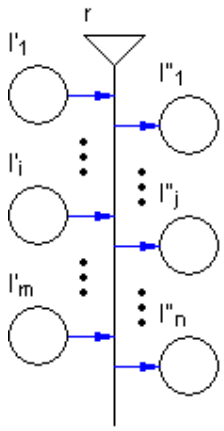


Figure 1: GN-transition

When the value of a type (calculated as a Boolean expression) is "true", the transition can become active, otherwise it cannot. The ordered four-tuple $E = \langle A, \pi_A, \pi_L, c, K, X, \Phi \rangle$ is called simplest reduced Generalized Net (briefly, we shall use again "GN") if:

a) A is a set of transitions;

b) π_A is a function giving the priorities of the transitions, i.e., $\pi_A : A \to N$, where $N = \{0, 1, 2,... \} \cup \{\infty\}$;

c) π_L is a function giving the priorities of the places, i.e., $\pi_L : L \to N$, where $L = pr_1 A \cup pr_2 A$, and $pr_i X$ is the i-th projection of the n-dimensional set, where $n \in N$; $n \ge 1$ and $1 \le k \le n$ (obviously, L is the set of all GN-places);

d) c is a function giving the capacities of the places, i.e., $c : L \to N$;

e) K is the set of the GN tokens;

f) X is the set of all initial characteristics, which the tokens can obtain when they enter the net;
g) Φ is a characteristic function which assigns new characteristics to every token when it transfers from an input to an output place of a given transition.

Over the generalized nets many types of operators are defined. One of these types is the set of hierarchical operators. One of them changes a given GN-place to a whole sub-net (see [2]).

The generalized nets [2] are an appropriate tool for modelling of real events phenomenon and complex structures because of their flexibility and universal modelling possibilities. They allow description of dynamic and steady–state behavior of complex systems. The main advantage of using the generalized nets is the possibility for fast simulation of various systems with complex structure. A change in the system functioning does not force a substantial change in the generalized net structure. They are a promising tool for describing and studying hybrid systems that are characterized as being concurrent, asynchronous, distributed, parallel and stochastic.

GN-model of storehouse processes

Models are executable encoding of knowledge and their execution generates solutions corresponding to the behaviors of the system being modelled [4].

A principal scheme of modelling the processes in storehouses by using generalized nets is further proposed. The developed GN-based model of storehouse processes is depicted on Figure 2. It contains three transitions and they can be divided (conditionally) in three contours, described respectively as:

- the commodities transfer,
- the requests transfer and control of their execution,
- the transport units transfer.

Three types of tokens: α, β and γ enter the net.

First, we shall give a particularly formal definition of the GN-model and then will comment on its functioning and the results of its work. We used the word "particular" because the forms of the transition condition predicates and the tokens characteristics will be described informally for the sake of easier text comprehension. However, in a program realization of the model they will be appropriately formalized.

The GN-transitions have the following forms:

$$Z_1 = \langle \{l_1, l_2, l_{15}, l_{17}\}, \{l_3, l_4, l_5\},$$

	l_3	l_4	l_5
l_1	*false*	$W_{1,4}$	*false*
l_2	*true*	*false*	*false*,
l_{15}	*false*	*false*	$W_{15,5}$
l_{17}	*false*	*false*	$W_{17,5}$

$$\wedge(\vee(l_1, l_3)), \vee(l_{15}, l_{17}))\rangle, \quad (2)$$

where

$W_{1,4}$ = $W_{2,4}$ = "there are free places in cells" & "there are free transport units for the commodities transfer",

$W_{15,5}$ = $W_{17,5}$ = "there are commodities for transfer".

The transition type shows that this transition is activated when there is at least one commodity and at least one free transport unit, simultaneously.

$$Z_2 = \langle \{l_4, l_5, l_6, l_{16}, l_{18}\}, \{l_6, l_7, l_8, l_{14}, l_{15}, l_{16}\},$$

	l_6	l_7	l_8	l_{14}	l_{15}	l_{16}
l_4	$W_{4,6}$	$W_{4,7}$	*false*	*false*	*false*	*false*
l_5	*false*	*false*	*false*	$W_{5,14}$	$W_{5,15}$	$W_{5,16}$
l_6	$W_{6,6}$	$W_{6,7}$	$W_{6,8}$	*false*	*false*	*false*,
l_{16}	*false*	*false*	*false*	$W_{16,14}$	$W_{16,15}$	$W_{16,16}$
l_{18}	*false*	*false*	*false*	$W_{18,14}$	$W_{18,15}$	$W_{18,16}$

$$\wedge(\vee(l_4, l_6)), \vee(l_5, l_{16}, l_{18}))\rangle, \quad (3)$$

where

$W_{4,7}$ = $W_{6,7}$ = "there is a request for a commodity",

$W_{4,6}$ = $W_{4,6}$ = $\neg W_{4,7}$; (where $\neg p$ is the negation of predicate p),

$W_{6,8}$ = "the commodity has expired",

$W_{5,14}$ = $W_{16,14}$ = $W_{18,14}$ = "there are commodities to be transferred from a cell to the exit" & "there is a free transport unit",

$W_{5,15}$ = $W_{16,15}$ = $W_{18.15}$ = "there are commodities to be transferred from the input to a cell" & "there is a free transport unit",

$W_{5,16}$ = $W_{16,16}$ = $W_{18,16}$ = "there are commodities to be transferred from a cell to a cell" & "there is a free transport unit".

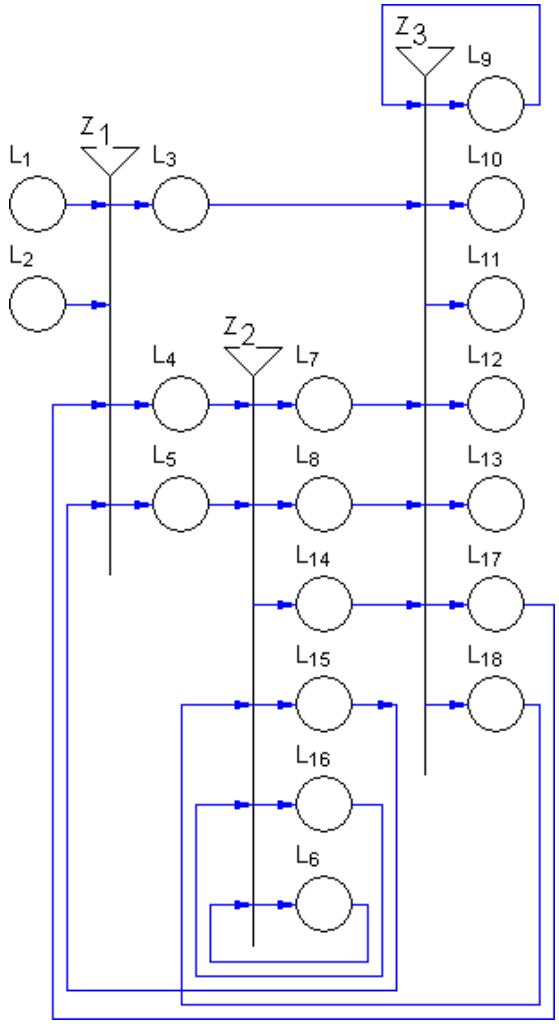


Figure 2: GN-model

$$Z_3 = \langle \{l_3, l_7, l_8, l_9, l_{14}\}, \{l_9, l_{10}, l_{11}, l_{12}, l_{13}, l_{17}, l_{18}\},$$

	l_9	l_{10}	l_{11}	l_{12}	l_{13}	l_{17}	l_{18}
l_3	*false*	*false*	*false*	$W_{3,12}$	$W_{3,13}$	*false*	*false*
l_7	$W_{7,9}$	$W_{7,10}$	$W_{7,11}$	*false*	*false*	*false*	*false*
l_8	*false*	*true*	*false*	*false*	*false*	*false*	*false*
l_9	$W_{9,9}$	$W_{9,10}$	$W_{9,11}$	*false*	*false*	*false*	*false*
l_{14}	*false*	*false*	*false*	*false*	*false*	$W_{14,17}$	$W_{14,18}$

$$\wedge (l_3, \vee(l_7, l_8, l_9)), l_{14} \rangle, \quad (4)$$

where

$W_{3,12}$ = “the request is executed”,

$W_{3,13} = \neg W_{3,12}$,

$W_{7,9} = W_{9,9}$ = “the commodity waits for transport units”,

$W_{7,10} = W_{9,10}$ = “the commodity has expired”,

$W_{7,11} = W_{9,11}$ = “the commodity is ready for discharging”,

$W_{14,\ 17}$ = “there are commodities to be transferred from the input to a cell”,

$W_{14,18}$ = “there are commodities to be transferred from a cell to a cell”.

The characteristic function Φ_i related to place l_i ($3 \le i \le 17$) determines the following token characteristics. The symbol in the brakes shows the argument of the function.

$\Phi_3[\gamma]$ – “time for waiting to receive the commodity”,

$\Phi_4[\alpha]$ – “time for the commodity to stay on the input”,

$\Phi_5[\beta]$ – “time for waiting for the transport unit at the last place, quantity of the transferred commodities”,

$\Phi_6[\alpha]$ – “number of the cell”,

$\Phi_7[\alpha]$, $\Phi_8[\alpha]$ – “time for the commodity to stay on the stocks”,

$\Phi_9[\alpha]$ – “ * ” (i.e. it is not defined),

$\Phi_{10}[\alpha]$ – “the commodity is scrapped, total time needed the commodity to stay in the storehouse”,

$\Phi_{11}[\alpha]$ – “the commodity is discharged, total time for the commodity to stay in the storehouse”,

$\Phi_{12}[\gamma]$ – “the request has been executed”,

$\Phi_{13}[\gamma]$ – “the request has not been executed”,

$\Phi_{14}[\beta]$, $\Phi_{15}[\beta]$ – “time to wait for the transport unit at the last place, quantity of the transferred commodities”,

$\Phi_{16}[\beta]$ – “quantity of the transferred commodities”,

$\Phi_{17}[\beta]$, $\Phi_{18}[\beta]$ – “time to wait for the transport unit at the last place, quantity of the transferred commodities”.

The different GN places have different priorities and capacities depending on their sense. The same is valid for the tokens and for the transitions. Transition Z_3 has the highest priority and Z_1 – the lowest one.

Transition Z_1: the highest priority has place l_1 (the input of the storehouse), after this – l_2 (the input of requests); places l_{15} and l_{17} are with equal priorities.

Transition Z_2: place l_4 has higher priority than place l_6, and places l_5, l_{16} and l_{18} are with equal priorities.

Transition Z_3: the priorities of places l_7, l_8 and l_9 satisfy the inequalities: $\pi_L(l_8) > \pi_L(l_9) > \pi_L(l_7)$, the other places have equal priorities. The place capacities and the token priorities are determined on the basis of the concrete storehouse parameters.

The GN functioning starts at a fixed time-moment according to a fixed time-scale (in relationship with the concrete storehouse) and it has a fixed duration.

In place l_1 α-tokens enter with initial characteristics: "*commodity name, cost, date of manufacturing, expiry date*". The β-tokens, which correspond to the transport units are located in places l_5, l_{14}, l_{15}, l_{16}, l_{17} and l_{18}, and they have as initial characteristics: "*transportation unit identification and parameters*". The γ-tokens, which enter place l_2 have as initial characteristics: "*number of the request, type of the necessary commodity*".

When commodities arrive at the storehouse, they are transported by the transport units to the cells (place l_6; via place l_4), if there are such units, and otherwise they wait in the storehouse input (l_1). The expired commodities are transferred to place l_{11} via place l_8 and these, for which there is a request – to place l_{10} via place l_7. The α-tokens, which correspond to the commodities, obtain a final characteristic: "*time for staying in the storehouse (and other parameters, which are interesting to the storehouse managers and dispatchers)*". The β-tokens, which correspond to the transport units, successively obtain the values of the transport expenses (and other parameters, which are interesting to the storehouse managers and dispatchers). The γ-tokens, which correspond to the commodity requests, obtain final characteristic: "*time for execution of the request (and other parameters, which are interesting to the storehouse managers and dispatchers)*".

In a result of the GN simulation of the storehouse processes, the user can get statistical data about the storehouse functioning.

Conclusions and future work

The above described GN-model of storehouse processes does not contain tools for optimization of the ordering of the commodities. Such tools could be described in the frames of another generalized net. There the basic criteria for commodity location will be realized. For example, the following factors for the functioning of the storehouses are very important:

- location of the commodities (this factor includes the solution of the problem, related to the disposition of the commodities on the cells and it can be formulated by the following: the cells from a low level must contain:
  - commodities with short expiry term;
  - commodities which will be soon discharged;
  - heavy commodities;
  - heterogeneous commodities which will be totally discharged);
- determination of the organization of the commodity streams and the criteria for the commodity discharging.

The above GN-based model could be further extended about the functioning of the transport units and about the order of receiving and execution of the different requests. The storehouse processes related to the last two object types could be described in the frames of other generalized nets that would be included in the above GN-model.

In the present form the GN-model is only on theoretical level, but it is applicable for real storehouse processes.

Acknowledgments

This work has been supported by the "Virtual Institute for Production Simulation Services" (SIM-SERV), project No. G7RT-CT-2001-05044.

References

[1] Banks J, Carson JS, and Nelson BL. *Discrete-Event System Simulation (2nd Edition)*. Prentice Hall, 1996.

[2] Atanassov K. *Generalized Nets*. World Scientific, Singapore, 1991.

[3] Murata T. *Petri Nets - Properties, Analysis and Applications*. Proceedings of the IEEE 1989; 77(4): 541-580.

[4] Leitch RR. *Modeling of complex dynamic systems*. IEE Proc., Part D 1987; 134(3): 245-250.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

ISTANBUL-ANKARA HIGH SPEED TRAIN TRAVEL ANALYSIS BY RAILWAY VEHICLE SIMULATION

Baha Güney, Imdat Taymaz*, Kemal Cakir and Emin Gündoğar
Engineering Faculty, Sakarya University,
Esentepe Campus, 54187, Sakarya, Turkey

Abstract

Railway administration wants to operate high speed trains with active tilting body aiming to shorten travel time and to provide safe passenger transportation between Istanbul-Ankara cities and to compete with motorway intercity bus transportation. It is demanded to settle a fleet, which runs on curves and slopes of the existing substructure. Travel with tilting body train sets takes shorter time than travel with conventional trains composed passenger coaches hauled by a locomotive. Main target is to improve travel time between Istanbul-Ankara by use of technical capabilities of these high speed tilting body train sets which provide extra speed in the curved railroad lines. A dynamic simulation model is developed in this study for the travel of the railway trains. Simulation has been made for different speed limits taking into account the technical data of train sets and locomotives and data including line parameters as curves and slopes. As a result of travel simulation, a comparison has been carried out between the performances of two different trains regarding their travel time.

Keywords: Dynamic simulation, railroad vehicle simulation

1. Introduction

Turkish State Railways Administration (TCDD) which takes about 4-5% share from total passenger transportation wants to operate high speed self powered train sets with active tilting body (tilting-train) for the purpose of performing fast and safe passenger transportation between Istanbul-Ankara and competing with other means of transportation [1]. It is demanded to settle a fleet of trains which can run on curves and slopes of the existing substructure in which travelling with tilting trains takes shorter time than travelling with conventional trains composed of railway passenger coaches hauled by locomotives (loco-train). The most important reason for demanding these high-speed tilting trains is to decrease the travelling time between Istanbul-Ankara line. Currently, the travel takes about 6-7 hours with loco-trains. This travel time is obtained by operating existing loco-trains with maximum speed of 140 km/h on existing non-rehabilitated railway lines. Travel time decrease can be realized both by running at higher speeds on the curved and straight sections. It is not possible to take full advantage of the high-speed capabilities of the trains on the line sections where there is also suburban train traffic because the average speed of the suburban trains is about 45-50 km/h, which brings speed limit to high-speed trains.

Therefore, it is planned to settle a train fleet which can decrease the travelling time between Istanbul-Ankara by operating tilting trains, that could run with a 30% more speed on the curved lines than loco-trains and reach a maximum speed of 230 km/h on straight lines. Alternative to this is to operate trains hauled by high-speed locomotives,

*Corresponding author. Phone:+90264 3460354
Fax: +90264 3460351, E-mail: taymaz@sakarya.edu.tr

by which the travelling time between Istanbul-Ankara can be reduced but not as much as those provided by tilting trains. However, both operating and maintaining loco-trains are more economical than tilting trains.

2. Theoretical Background

2.1 Factors Effecting the Travelling Time

The curved lines has a track cant which is constructed by lowering the ground level of the inside rail to avoid vehicle derailing outside the line with the effect of the centrifugal force trying to push the vehicle outside the curve [2]. The need for track cant can be calculated as shown in equation 1.

$$d = 8\frac{v^2}{R} \qquad (1)$$

where;
d: Track cant, mm
v: Speed of train, km/hour
R: Rradius of curvature, m

The track cant couldn't be beyond a limit value because, when a train stops in the curve, it should not derail inside the curve because of gravity force. Therefore, the train speed is limited by the curve radius and track cant together. The maximum track cant for the curved lines in this route is 130 mm. It is not possible to operate the tilting and loco-trains with the desired high speeds because of many curved sections on the existing route. Besides, for a comfortable travel, the lateral acceleration affecting a seated passenger has to be less than $\gamma \leq 0.65 m/s^2$. γ value can be calculated as follows.

$$\gamma = \frac{\upsilon^2}{R} - g.\tan\alpha \qquad (2)$$

where;
g: Gravitational acceleration, m/s^2
α: Track cant angle
v: Speed of train, km/hour
R: Radius of curvature, m

In order to solve speed limit problem in curved lines, there is a mechanism, which tilts the railway vehicle body inside the curve. In this design, which is called as active tilting system, railway vehicle is tilted on the bogies. Therefore, centre of gravity is shifted to curve centre side, which balances centrifugal force increase resulting from increased speed. When these vehicles are compared with conventional vehicles, it is experienced that the trains with tilting system is capable of running in curved lines with 30% more speed than the others. Tilting mechanism is effective at vehicle end points on bogies. This mechanism can be hydrostatically or electro-mechanically actuating. In Figure-1, a sample carbody tilting mechanism is shown.

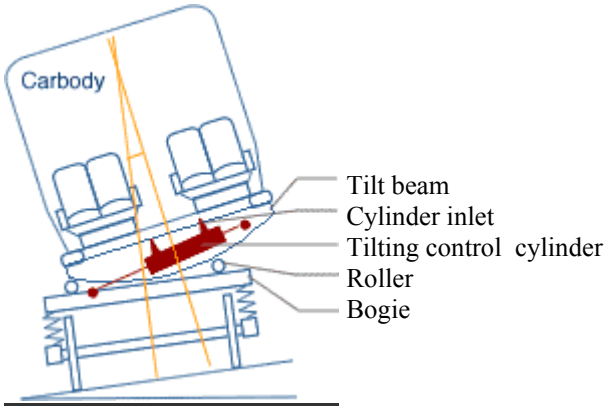


Figure-1 Sample carbody tilting mechanism

2.2 Properties of Tilting Trains

Maximum speed of trains with tilting body is about 200-230 km/h. These tilting trains are not actually high-speed trains according to definition of European Union of Railways [3]. However, since the speeds of trains under investigation in this study are higher than those currently operating conventional trains, it is named as high speed. The tilting system has been developed to increase the operational speed of the trains on lines which have many curves. With the system, car bodies are tilted at curves to compensate for unbalanced carbody centrifugal acceleration to a greater extent than the compensation produced by the track cant, so that passengers do not feel centrifugal acceleration and thus trains can run at higher speed at curves. Tilting trains are used in the European countries like Italy, Sweden, and Spain, which have curved railway lines. There are tilting trains both electrically or diesel powered.

2.3 Traction Forces

Traction force is the force applied by the locomotive to pull the train. Figure 2 shows the traction-force vs. speed diagram used in the simulation of tilting train composed of 7 vehicles. Train maximum traction power is 4000kW. As seen in Figure-2, traction force is constant up to 80 km/h which means that power increases to its maximum value. Above 80 km/h, the power is constant at 4000kW but the traction force decreases proportionally as the speed increases since power is the product of force and speed. Traction-force versus speed diagram for a 5000kW power electric locomotive, which is used for the loco-train simulation, has got almost similar shaped diagram but in fact rating is different.

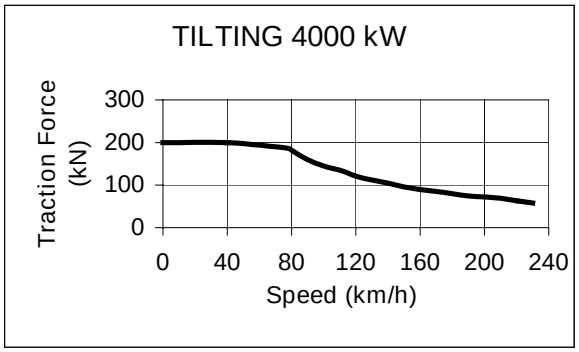


Figure-2. Traction force vs. speed diagram for tilting train.

2.4 Resistance Forces

The total resistance force resists to the train's movement. Equation 3 given below is suggested by the vehicle manufacturers to calculate the total value of the resistance forces (R_T) that acts against the movement of tilting trains.

$$R_T = \left[f_{aA} \cdot \frac{n_{aA}}{n_{gA}} \cdot m_g + f_{nA} \cdot \frac{n_{nA}}{n_{gA}} \cdot m_g \right] \cdot g + f_1 \cdot \left(f_2 + n_w \cdot f_3 \right) \cdot \sigma_{air} \cdot \frac{\left(v + v_g \right)}{3.6} + \left[\left(n_{SW} \cdot c_{wSW} + n_{TW} \cdot c_{wTW} + n_{MW} \cdot c_{wMW} \right) \cdot A \cdot \frac{\sigma_{air}}{2} \cdot \frac{\left(v + v_g \right)^2}{3.6^2} \right] \quad (3)$$

The variables, constants and the values assigned to them in the simulation model are shown in Table1.

Table 1: The variables and constants for the resistance forces affecting tilting train

Variables & Constants	Assigned Values	Definition
f_{aA}	0.9	Power axle factor
f_{nA}	0.6	Driven axle factor
n_{aA}	8	Power axle amount
n_{nA}	20	Driven axle quantity
n_{gA}	28	Total axle quantity
m_g	392	Train total mass, ton
g	9.81	Gravitational acceleration, m/s²
f_1	0.9	Correction factor
f_2	45	Traction factor
f_3	3.2	Air conditioning factor
n_w	7	Total # of vehicles
σ_{air}	1.225	Air density, kg/m³
v	-	Speed, km/hour
v_g	15	Opposite wind speed, km/hour
n_{SW}	2	Number of motor cars
n_{TW}	2	Transformer vehicles
n_{MW}	3	Trailer railway cars
c_{wSW}	0.4	Motor car rolling factor
c_{wTW}	0.12	Transformer car rolling factor
c_{wMW}	0.105	Trailer car rolling factor
A	10	Car end cross-section area, m²

For loco-train configuration, the total resistance force equation is given as follows [4].

$$R_T = R_A + R_K + R_R + R_Y \quad (4)$$

The components of the total resistance equation are given below:

R_A: Aerodynamic resistance force, N

$$R_A = \frac{1}{2} c_w A \left(\frac{v + 12}{10} \right)^2 \quad (5)$$

R_K : Curve resistance force, N

$$R_K = \frac{500}{R} m_g g \quad (6)$$

R_R : Slope resistance force, N

$$R_R = s\, m_g\, g \quad (7)$$

R_Y : Rolling resistance force, N

$$R_Y = (c_d m_d + c_r m_r) g \quad (8)$$

The variables and constants in equations 4 - 8 are given in Table 2.

Table 2: The variables and constants for the resistance forces affecting loco-train.

Variables Constants	Appointed values	Definition
c_w	2.1	Train air resistance factor
A	10.5	Front face cross-section area of vehicle
v	-	Speed of vehicle, km/hour
R	-	Radius of curve, m
s	-	Slope, ‰
m_g	419	Weight of train, ton
g	9.81	Gravitational acceleration, m/s²
c_d	2.5	Powered Axle revolving factor
m_d	90	Total axle load of powered axles, ton
c_r	2	Revolving factor of non actuated axles
m_r	329	Total weight of non actuated axles
v_g	-	Opposite wind speed, km/h

There are speed limit regulations by railway administrations because of the line curves. The speed limitations are used in the simulation program to restrict the speed of the train running in the curves. Tilting trains are allowed to run at a speed 30 percent more than those of loco-trains.

The difference between traction and resistance forces will result in either positive or negative acceleration causing slowing down or speeding up. Acceleration or deceleration is calculated by dividing this difference to total train weight. If the deceleration is less than the desired slowing, brakes will be operated to provide extra resistance to the motion.

3. Simulation Model

Istanbul-Ankara route train travel simulation is designed as time incremented continuous simulation model [5]. The code of the program is built in Visual Basic programming language. There are two program modules, first is for data input and data organization and second is for simulation run.

The data of distance-slope, distance-curve, distance-speed restrictions, that are directly read from the railroad technical map [6], arranged together in a database and made ready for use by the second module where the dynamic simulation is run. To make data input easy only the slope starting point and slope amount and length of slope are entered. All curves are entered into database with starting and end points together with related curve radius. The speed of train is limited while the train is passing through the stations. The train may make a planned stop at a station or pass the station without stopping but with a limited speed. Therefore, destination of the station, length of the station from start to end and whether the train will stop or not at the specific station are all recorded to the database. The first module arranges the input raw data for each unit length of the railway line and records limiting maximum speed because of curves, slopes, stations or crossings.

Three main events occur during dynamic simulation depending on the current speed of the train and actual position on the line. These events are, acceleration, constant speed and slowing down events. There are traction forces, resistance forces and braking forces affecting the vehicles' motion. During the acceleration, the vehicle makes use of the traction force calculated from the traction force-velocity diagram. The difference between traction force and total resistance force is net force to accelerate the train. As speed increases, net force continuously decreases until it reaches a balance point where traction force is just equal to total resistance force. Whenever there is a positive net force, train is capable of accelerating, that is to speed up.

Dynamic simulation module consists of program sections that provide speeding up, constant speed and slowing down with a main control program section, which puts either of them into execution. The time increment of continuous simulation is 0,01 seconds during which the train will not travel more than the distance increment of the organized line data. The simulation scheme is shown in Figure-3.

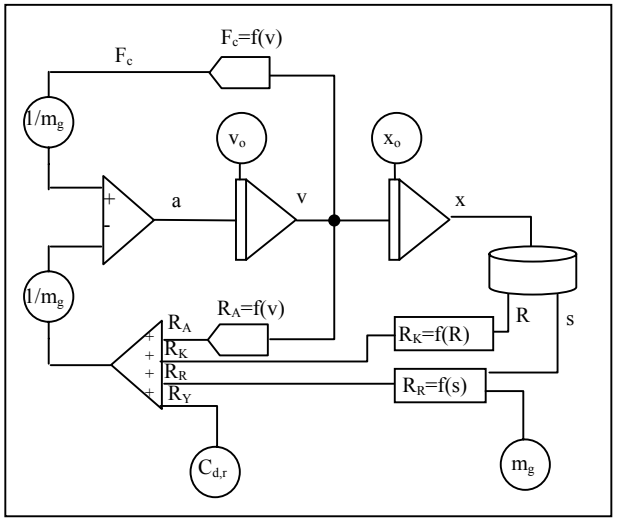


Figure -3. Simulation scheme.

The acceleration obtained by net force summer can be used to calculate the speed and the distance. The records like, slope value, curve radius can be read from the database corresponding to the distance travelled. For the calculation of speed and distance, integrators use the following equations,

$$v = \int_{t}^{t+\Delta t} a dt + v_0 \tag{9}$$

$$x = \int_{t}^{t+\Delta t} v dt + x_0 \tag{10}$$

where: a, v and x are acceleration, velocity and distance respectively. Vehicle will accelerate by making use of maximum of the traction forces, if there is no speed restriction. Technically, vehicle will reach the maximum speed in this condition. When the vehicle reaches the maximum limited speed on the line, by decreasing the traction forces until the level that will just balance the resistance forces, vehicle travels with a constant speed. This is similar to reducing the traction level in the actual case. This situation is realized in the constant speed module of the dynamic simulation program.

The deceleration module of the program will be active while the vehicle is moving with a certain speed, there will be a lower speed restriction than it's actual speed in the forward section of the line. This speed limitation can occur because of the curves, stations and crossing or planned station stops. For these coming events, program will decrease its speed continuously to the limit value without exceeding the allowed deceleration level by making forward simulations in every step. For this purpose, the current speed and the first speed limit at a distance ahead of train are compared and the distance to decelerate to that limit speed without exceeding the deceleration comfort limit is calculated. Whenever remaining distance is equal to that calculated safe decelerating distance, train starts reducing its speed until it reaches the speed-limiting zone. At constant limit speed train travels until speed-limiting distance is passed. Then, further conditions of the line are taken into execution by one of the three the program module.

4. Results and Discussion

The first module organized the speed restrictions of the line, by organizing the database. The integrity and validity of these data are examined by comparing them with the real line values. Program was operated for one-way direction between Istanbul-Ankara route for tilting trains and loco-trains. At Bostancı, Izmit, Arifiye and Eskişehir stations two minutes planned stoppages have been realized. The station entry speed for every station is limited to 70km/h and the station switch exit speed is limited to 90km/h. Between Haydarpaşa-Gebze and Sincan-Ankara suburban areas, only 120km/h maximum speed is allowed due to safety regulations. For the speed restrictions and stoppages, the program has been operated for the decelerations 0.5, 0.7 and 1 m/s². The reason for choosing these different decelerations is the passenger comfort. The deceleration, 1m/s², resulting in worst comfort is selected for the calculation of minimum travelling time.

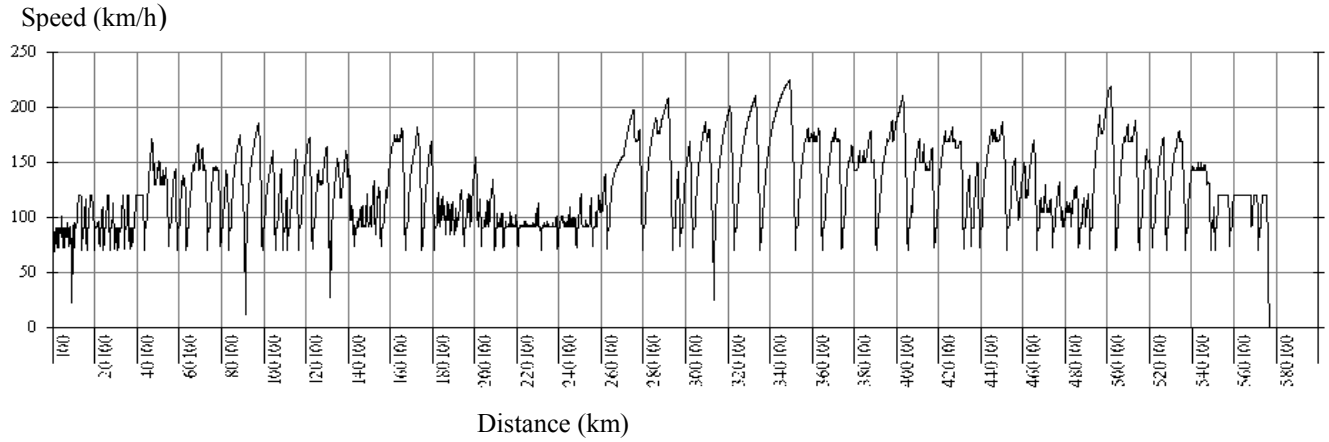


Figure 4. Speed versus distance diagram for Istanbul –Ankara railway line (576 km.)

Simulation resulted in following travel times for different decelerations. (Table 3)

Table 3: Tilting train travel time

Deceleration	0.5 m/s²	0.7 m/s²	1 m/s²
Travelling Time (hour: minute)	5:03	4:58	4:54

The simulation program has run for the loco-train with the same decelerations in the same route line and in Table 4, the results are given.

Table 4: Loco-train travel time

Deceleration	0.5 m/s²	0.7 m/s²	1 m/s²
Travelling Time (hour: minute)	5:38	5:34	5:30

Speed- distance diagram of tilting body train between Istanbul – Ankara line for the maximum deceleration level of $0.7 m/s^2$ is shown in Figure 4. The average speed for the tilting train is about 117 km/hour and for the loco-train is 104 km/hour. When the total travelling times are examined, it is seen that the tilting train has travelled the route in 4 hours 58 minutes and the loco-train has travelled the same route in 5 hours 34 minutes, which is 36 minutes more than the other, which corresponds to nearly %12 more travelling time for the loco-train.

The railway administration requires a maximum 2 hours 28 minutes travelling time for the 236 km section between Gebze-Inönü which is the most problematic part of the total line between Istanbul-Ankara because of the narrow curves and high slopes. The loco-train travels this line section in 2 hours 31 minutes under $0.7 m/s^2$ deceleration condition, therefore almost satisfies this requirement.

The difference in the travelling times is because of the capability of the tilting train to travel with a 30% more speed capability than loco-trains in the curves. In the line sections where the curves are dense, the tilting train has performed better than other as expected. The result of rehabilitation works on the lines in the parts with narrow curves will affect both of the alternatives in the same ratio. Therefore, the line rehabilitations will keep same percentage difference in travelling time although travel time will decrease for both of them.

5. Conclusion

Dynamic simulation study for railway vehicles has provided comparison of performances of different trains on the railway lines under investigation. Therefore, valuable information is obtained by the railway administration before making investment to new fleet purchasing and operating decisions. Besides, various scenarios such as different train configuration, other planned stops at stations and run on different line routes can be designed to see the performance of a train and to make preliminary schedule of the arrival times to destinations.

6. References

[1] TCDD. *Specifications and appendices for high-speed train supply tender, 2001,Ankara*

[2] Hay WW. *Railroad engineering.* Canada: John Wiley, 1982, Pg. 69-89.

[3] UIC Code 660 OR, *Measures to ensure the technical compatibility of high speed trains*, 2nd Edition, August 2002.

[4] Yetiş N, Güney B, Çakır K. *Rail vehicle Simulation*, Proceedings of Advances in Simulation Symposium, Istanbul, Editors: Kaylan, A.R., Ören, T., July 1992, Pg.83-94

[5] Matko D, Karba R, Zupanic B. *Simulation and modelling of continuous systems.* New York: P. Hall, 1992.

[6] *TCDD. Ankara-Istanbul* Railroad *line curve and slope versus distance diagrams.*

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

MODELLING OF A ONE PASS SMOKE TUBE BOILER

Claus M. S. Karstensen

Aalborg Industries A/S, Gasværksvej 24, P.O. Box 661, DK-9100 Aalborg, Denmark
Tel: +45 99 30 44 47, Fax: +45 99 30 44 53, cka@aalborg-industries.com, http//www.aalborg-industries.com/

Kim Sørensen

Aalborg University, Institute of Energy Technology, Pontoppidanstræde 101, DK - 9220 Aalborg, Denmark
Tel: +45 96 35 92 48, Fax: +45 98 15 14 11, kso@iet.auc.dk, http://www.iet.auc.dk/
Aalborg Industries A/S, Gasværksvej 24, P.O. Box 661, DK-9100 Aalborg, Denmark
Tel: +45 99 30 45 43, Fax: +45 99 30 44 53, kso@aalborg-industries.com, http//www.aalborg-industries.com/

Abstract

A nonlinear state-space model with five states describing a one pass smoke tube boiler has been formulated. By means of mass- and energy-balance the model describes the dynamics of the Furnace, the Convection Zone and the Water/Steam Part and the three sub models are merged into an overall model. The model is further linearized for use in a linear control design. The simulations have been carried out by means of MATLAB/SIMULINK and the models have been verified with measurements from a full scale boiler plant. Parameters in the model that are difficult to calculate have been estimated and the method used is the Prediction Error Method.

Keywords: Boiler dynamic, nonlinear modeling, Simulation, Parameter estimation and Simulink.

Nomenclature list

c_p	$\frac{J}{kg \cdot K}$	specific heat capacity
$\dot{m}$	$\frac{kg}{s}$	mass flow
$\dot{q}$	$\frac{J}{s}$	energy flow
H_u	$\frac{J}{kg}$	calorific value
M	kg	mass
U	J	internal energy
α	$\frac{J}{s \cdot m^2 \cdot K}$	coefficient of heat transfer
v	$\frac{m^3}{kg}$	specific volume
h	$\frac{J}{kg}$	Enthalpy
P	Pa	Pressure
L	m	Level
T	K	Temperature
V	m^3	Volume
α, β		Parameters
ρ	$\frac{kg}{m^3}$	Density
A	m^2	Area

Subscripts

b	bubble
fp	flue gas pipe
fw	feed water
fn	furnace
r	radiation
c	convection
s	steam
w	water

1. Introduction

A dynamic model of the boiler covering the furnace-/convection and the water-/steam part has been formulated. In order to make a dynamic model of the boiler, the input and output of the system must be identified. Further, the boiler is decomposed into three sub models consisting of the furnace, the convection part and the water-/steam space. Finally, the three models are collected to form the dynamic model of the boiler. The most of the model is described by using mass- and energy balance. This is done since most of the boiler dynamics are captured in these general equations. Simulation of the model has been carried out by Matlab/Simulink. For verification of the model, testing has been carried out on a full scale test plant where measured output data are compared with the simulated output. Parameters which are difficult to calculate are estimated by minimize the prediction error. To find the optimum, Gauss-Newton algorithm is used.

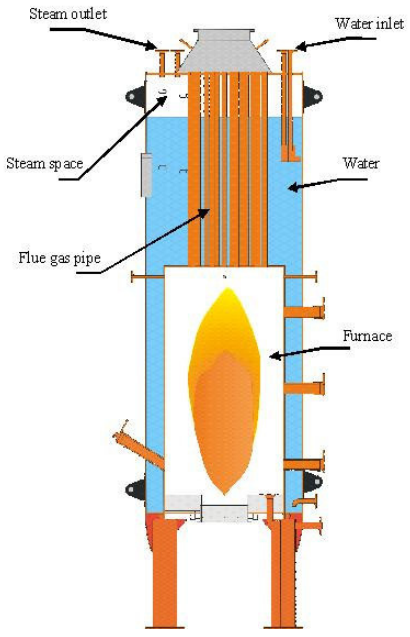


Figure 2.1: Principle of the boiler type

2. Inputs and Outputs

Before the system is decomposed input and output of the model are identified. The overall goal is to control the *water level* and the *steam pressure* more efficiently and the *controllable* inputs to the boiler are the amount of *feed water* and *fuel/combustion air*. The fuel/air ratio is mechanically fixed, which means that, from a control point of view, the system is a two inputs and two outputs system.

The mentioned two inputs are incomplete to describe the model satisfactory. It is therefore necessary to expand the number of inputs to give more specific information. The mass flows entering the system are *fuel*, *air* and *feed water* and the mass flows out of the system are *steam* and *flue gas*. All the mass flows enter and leave the system with a certain temperature, and together with the mass flows this express the amount of energy transported through the system. If no losses through the walls are considered the global mass and energy balance can be derived. Figure 2.2 illustrate all input and outputs.

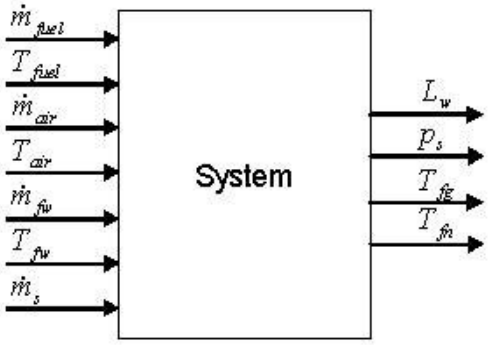


Figure 2.2: Inputs and outputs to the boiler model.

As illustrated on figure 2.2 the temperature from the *furnace* and the *flue gas,* are simulated. These two signals are only used to calculated the heat transfer from the furnace to the water and from the convection part to the water, and thereby have a more precisely estimation of the individually heat transfer coefficients. It shall also be mentioned that the mass flow of the steam is used as an input to the model even it is an output of the boiler.

3. Sub-system models

In order to simplify the modeling process the boiler is decomposed into 3 sub models: one contain the *furnace*, one contain the *convection part* and one contain the *water/steam part*. Figure 3.1 shows the interaction between the different sub-systems.

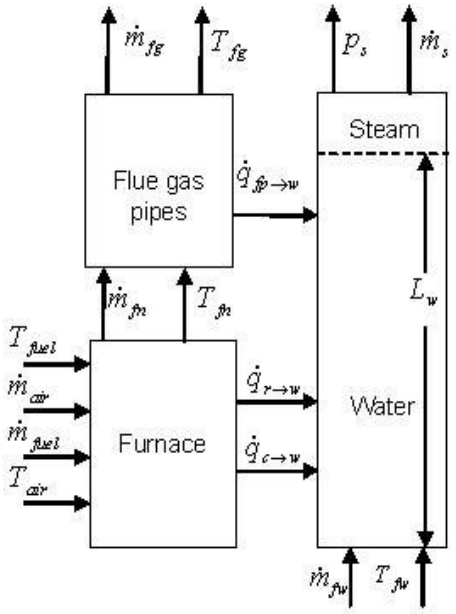


Figure 3.1: Sub-systems of the boiler.

4. Modelling

When modelling the process the following assumptions are made:

- Water and steam is always in a saturated state.
- Metal parts in the furnace, convection part and the water/steam space have the same temperature as water and steam.
- The heat transfer from the convection parts to the steam is negliciated.
- All losses through the walls of the boiler are negliciated

4.1 Furnace

The dynamic behavior the air in the furnace is captured by the following energy balance equation:

$$\dot{q}_{fuel} + \dot{q}_{air} - \dot{q}_{fn} - \dot{q}_{r\to w} - \dot{q}_{c\to w} - \frac{dU_{fn}}{dt} = 0 \quad (4.1.1)$$

where each component can be written as:

$$U_{fn} = M_{fn} \cdot h_{fn} = V_{fn} \cdot \rho_{fn} \cdot c_{p,fn} \cdot \overline{T}_{fn} \quad (4.1.2)$$

$$\dot{q}_{fuel} = \dot{m}_{fuel} \cdot \left(H_{fuel} + h_{fuel}\right) = \dot{m}_{fuel} \cdot \left(H_{fuel} + c_{p,fuel} \cdot T_{fuel}\right) \quad (4.1.3)$$

$$\dot{q}_{air} = \dot{m}_{air} \cdot h_{air} = \dot{m}_{air} \cdot c_{p,air} \cdot T_{air} \quad (4.1.4)$$

$$\dot{q}_{fn} = \dot{m}_{fn} \cdot h_{fn} = \dot{m}_{fn} \cdot c_{p,fn} \cdot T_{fn} \quad (4.1.5)$$

$$\dot{q}_{r\to w} = A_{fn} \cdot \alpha_r \cdot \left(\overline{T}_{fn}^{\,4} - T_w^4\right) \quad (4.1.6)$$

$$\dot{m}_{combustion} = \dot{m}_{fuel} + \dot{m}_{air} \quad (4.1.7)$$

$$\dot{q}_{c\to w} = A_{fn} \cdot \left(\frac{\dot{m}_{air}}{\dot{m}_{air,rel}}\right)^{0.8} \cdot \alpha_c \cdot \left(\overline{T}_{fn} - T_w\right) \quad (4.1.8)$$

$$\overline{T}_{fn} = \left(T_{fn} + T_{\text{combustion}}\right) \cdot \tfrac{1}{2} \quad (4.1.9)$$

$$T_{\text{combustion}} = \frac{\left(\dot{q}_{fuel} + \dot{q}_{\text{air}}\right)}{\dot{m}_{combustion} \cdot c_{p,combustion}} \quad (4.1.10)$$

By rearranging equation (4.1.1) and insert equation (4.1.2) the following expression can be derived:

$$\frac{d\overline{T}_{\text{fn}}}{dt} = \frac{\left(\dot{q}_{fuel} + \dot{q}_{\text{air}} - \dot{q}_{\text{r}\to\text{w}} - \dot{q}_{\text{c}\to\text{w}} - \dot{q}_{\text{fp}}\right)}{V_{fn} \cdot c_{p,fn} \cdot \rho_{fn}} \quad (4.1.11)$$

4.2 Convection part

The dynamic behavior of the air in the convection part is captured by the following energy balance equation:

$$\dot{q}_{fn} - \dot{q}_{fg} - \dot{q}_{fg\to w} - \frac{dU_{fg}}{dt} = 0 \quad (4.2.1)$$

Each component can be written as:

4.2.2) $$U_{fg} = M_{fg} \cdot h_{fg} = M_{fg} \cdot c_{p,fg} \cdot \overline{T}_{fg}$$

4.2.3) $$\overline{T}_{fg} = \left(T_{fn} + T_{fg}\right) \cdot \tfrac{1}{2}$$

4.2.4) $$\dot{q}_{fg} = \dot{m}_{fg} \cdot h_{fg} = \dot{m}_{fg} \cdot c_{p,fg} \cdot T_{fg}$$

4.2.5) $$\dot{m}_{fg} = \dot{m}_{fn}$$

4.2.6) $$\dot{q}_{fg \to w} = A_{fg} \cdot \left(\frac{\dot{m}_{air}}{\dot{m}_{air,rel}} \right)^{0.8} \cdot \alpha_{fg} \cdot \overline{T}_{fg}$$

4.2.7) $$M_{fg} = V_{fg} \cdot \rho_{fg}$$

Inserting equation (4.2.1) into (4.2.2) yields:

4.2.8) $$\frac{d\overline{T}_{fp}}{dt} = \frac{\dot{q}_{fn} - \dot{q}_{fp} - \dot{q}_{fp \to w}}{V_{fp} \cdot \rho_{fp} \cdot c_{p,fp}}$$

4.3 Water/steam part

To model the dynamic behavior of the system figure 4.1 illustrate the interaction of mass and energy flows between the *furnace/convection part* the *feed water*, the *water* and the *steam*.

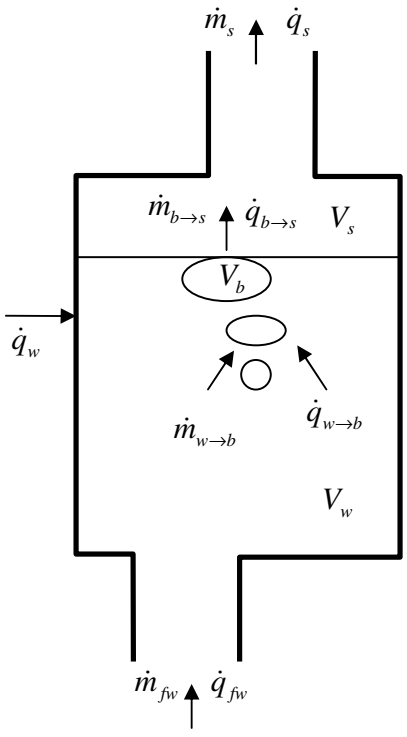


Figure 4.1: Model of the water/steam space.

The total volume of the water and steam in the boiler is given as:

4.3.1) $$V_t = V_w + V_s + V_b$$

The total mass balance for the water/steam is given as:

4.3.2) $$\frac{d}{dt}\left(\rho_s\left(V_s + V_b\right) + \rho_w V_w\right) = \dot{m}_{fw} - \dot{m}_s$$

differentiation (using chain rules) and rearranging the equation results in the following expression:

4.3.3) $$\frac{dP_s}{\partial t} \underbrace{\left(\frac{d\rho_s}{dP_s}\left(V_t - V_w\right) + \frac{d\rho_w}{dP_s} V_w \right)}_{A_{11}} + \frac{dV_w}{dt} \left(\underbrace{\rho_w - \rho_s}_{A_{12}} \right) = \underbrace{\dot{m}_{fw} - \dot{m}_s}_{B_1}$$

The total energy balance for the water/steam space is given as:

4.3.4) $$\frac{d}{dt}\left(U_w + U_b + U_s + U_m\right) = \dot{q}_w + \dot{q}_{fw} - \dot{q}_s$$

i.e

4.3.5)

$$\frac{d}{dt}\left(\begin{matrix} \rho_w V_w \left(h_w - v_w P_s\right) + \rho_s \left(V_b + V_s\right) \\ \left(h_s - v_s P_s\right) + \rho_m V_m c_{p,m} T_s \end{matrix} \right) = \dot{q}_w + \dot{q}_{fw} - \dot{q}_s$$

where

4.3.6) $$\dot{q}_w = \dot{q}_{r \to w} + \dot{q}_{c \to w} + \dot{q}_{fg \to w}$$

Differentiating (using chain rules) and rearrange equation 4.3.5 can be written as:

4.3.7)

$$\frac{dP_s}{dt}\underbrace{\left(\begin{array}{l}\frac{dh_w}{dP_s}\rho_w V_w+\frac{d\rho_w}{dP_s}h_w V_w+\\ \frac{dh_s}{dP_s}\rho_s\left(V_t-V_w\right)+\frac{d\rho_s}{dP_s}h_s\left(V_t-V_w\right)-\\ V_t+\frac{dT_s}{dP_s}\rho_m V_m c_{p,m}\end{array}\right)}_{A_{21}}+\frac{dV_w}{dt}\underbrace{\left(h_w\rho_w-h_s\rho_s\right)}_{A_{22}}=\underbrace{\dot{q}_w+\dot{q}_{fw}-\dot{q}_s}_{B_2}$$

The mass balance for the water and steam under the water level is [7]:

4.3.8) $$\frac{d}{dt}\left(\rho_w V_w+\rho_s V_b\right)=\dot{m}_{fw}-\dot{m}_{b\to s}$$

An expression to describe the flow $\dot{m}_{b\to s}$ is needed. For this purpose the mass balance for the bubble in the water is expressed as:

4.3.9) $$\begin{aligned}&\frac{d\left(\rho_s V_b\right)}{dt}=\dot{m}_{w\to b}-\dot{m}_{b\to s}\ \Leftrightarrow\\ &\dot{m}_{w\to b}=\frac{d\left(\rho_s V_b\right)}{dt}+\dot{m}_{b\to s}\end{aligned}$$

The mass flow of evaporation from the water to the steam space is described as an empirical equation [7] given as:

4.3.10) $$\dot{m}_{b\to s}=\alpha\frac{V_b}{V_w}+\beta\dot{m}_{w\to b}$$

Combining equation 4.3.9 and equation 4.3.10 and isolating $\dot{m}_{b\to s}$ gives:

4.3.11) $$\dot{m}_{b\to s}=\frac{\alpha\frac{V_b}{V_w}+\beta\frac{d\left(\rho_s V_b\right)}{dt}}{1-\beta}$$

Inserting equation 4.3.11 in equation 4.3.8 and after differentiation (using the chain rules) the equations can be written as:

$$\underbrace{\frac{dP_s}{dt}\left(\left(1-\beta\right)V_w\frac{d\rho_w}{dP}+V_b\frac{d\rho_s}{dP}\right)}_{A_{31}}+$$

4.3.12) $$\underbrace{\frac{dV_w}{dt}\left(1-\beta\right)\rho_w}_{A_{32}}+\frac{dV_b}{dt}\underbrace{\rho_s}_{A_{33}}=\underbrace{\left(1-\beta\right)\dot{m}_{fw}-\alpha\frac{V_b}{V_w}}_{B_3}$$

Combining equation 4.3.3, 4.3.7 & 4.3.12 yields:

$$A\cdot y'=B\Leftrightarrow\begin{bmatrix}A_{11}&A_{12}&0\\A_{21}&A_{22}&0\\A_{31}&A_{32}&A_{33}\end{bmatrix}\cdot\begin{bmatrix}\frac{dP_s}{dt}\\\frac{dV_w}{dt}\\\frac{dV_b}{dt}\end{bmatrix}=\begin{bmatrix}B_1\\B_2\\B_3\end{bmatrix}$$

which is an ordinary set of differential equations (ODE).

5. Parameter estimation

In the model there are five parameters which are not well determined using theoretical methods. These are determined using Prediction Error Method [6] and the performance function $F(\theta)$ is defined as:

$$F(\theta)=\frac{1}{n}\sum_{t=11}^{n}\left(\varepsilon(t,\theta)\right)^2$$

$$\varepsilon(t,\theta)=y(t)-\hat{y}(t,\theta)$$

where $\hat{y}(t,\theta)$ is the predicted output from the model dependent of the parameter vector θ, y is the measured output from the plant and $\varepsilon(t,\theta)$ is the prediction error. The algorithm used to find the minimum of the prediction error is Gauss Newton [6]. The parameters which are estimated are α_c, α_r, α_{fp}, α and β. The parameter estimation was based on experimental measurements on a full

scale 3 ton/h plant. The measurements were carried out in open loop where regulators for water level control and pressure control were removed. The load was 50 percent with pressure at 7 bars. All estimated parameters fit very well with calculated values, see [2] and [10]. The results are seen in table 5.1.

$\alpha_c \left[J/m^2K\right]$	22.14
$\alpha_r \left[J/m^2K^4\right]$	5.17e-9
$\alpha_{fp} \left[J/m^2K\right]$	115.65
$\alpha \ [kg]$	2.93
$\beta[\cdot]$	0.87

Table 5.1: Values of estimates coefficients

6. Verification of the nonlinear model

Verification of the nonlinear model is carried out at 50 percent load with pressure at 7 bars. The results are seen in figure 6,1, 6.2 and 6.3. The measurements seen in figure 6.1 are also used to estimate the parameters listed in table 5.1. The results in figure 6.1, 6.2 and 6.3 show that there is very good agreement between the model and the experimental data. Note that shrink and swell effect is captured very well by the model.

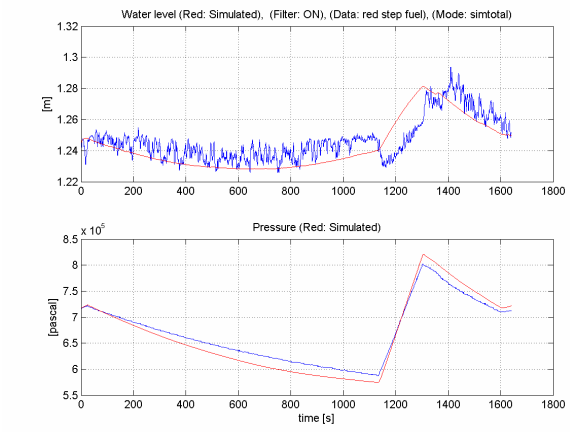


Figure 6.1: Step response on fuel flow

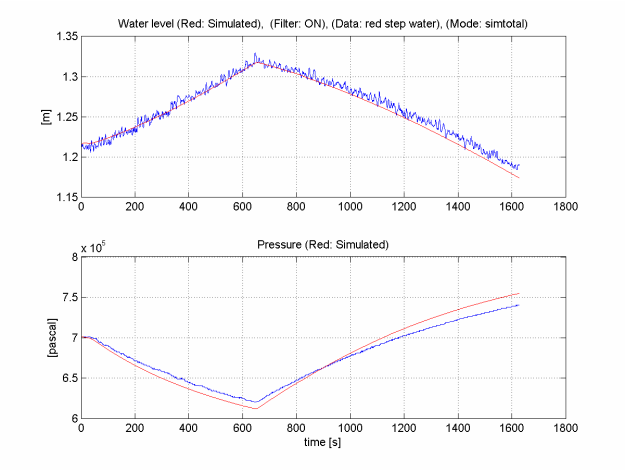


Figure 6.2: Step response on feed water flow

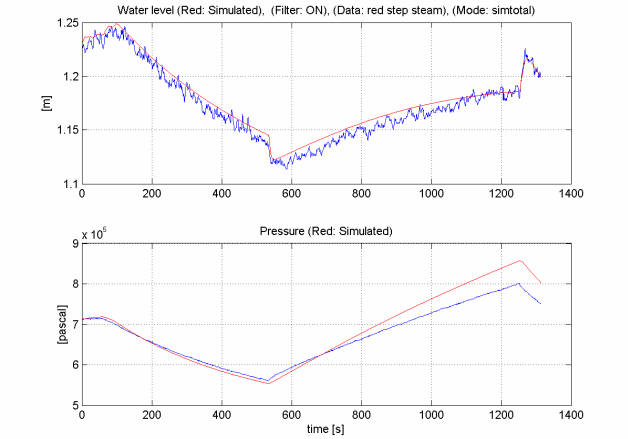


Figure 6.3: Step response on steam flow

7. Linear model

To construct a linear controller it is necessary to linearize the model. The principle used is to calculate the terms of a Tayler-series expansion and then insert the operating point in the equation. The principle is illustrated as:

$$f(x,y) \approx f(\bar{x},\bar{y}) + \left.\frac{df(x,y)}{dx}\right|_{\substack{x=\bar{x},\\ y=\bar{y}}} \cdot \hat{x} + \left.\frac{df(x,y)}{dy}\right|_{\substack{x=\bar{x},\\ y=\bar{y}}} \cdot \hat{y}$$

where $\hat{x} = x - \bar{x}$ and $\hat{y} = y - \bar{y}$ are the small-signals values, and $\bar{x}$ and $\bar{y}$ are the operating points.

The operating points for the variables in the models which have to be found are: $T_{fn}, T_{fg}, T_{air}, T_{fuel}, T_{fw}, P_s, V_w, V_b, \dot{m}_{fuel}, \dot{m}_{air}, \dot{m}_{fw}$ and $\dot{m}_s$. P_s and $\dot{m}_{fw}$ can be chosen to the desired operating point. When L_w also is chosen then V_w and V_b can be calculated by equation 4.3.12 utilizing that $V_b = L_w A_w + V_0 - V_w$. In steady state $\dot{m}_s = \dot{m}_{fw}$. Using the fact that there is a fixed ratio between $\dot{m}_{air}$ and $\dot{m}_{fuel}$, $\dot{m}_{air}$ can be removed. T_{air}, T_{fuel} and T_{fw} are regarded as disturbances and known from measurements. Finally there are three unknown variables (T_{fn}, T_{fg} and $\dot{m}_{fuel}$) and three equations (4.1.11, 4.2.8 & 4.3.7). These operating points were calculated by the MATLAB function *trim.*

The linear model is formulated as a state space model and given by equation 7.1:

$$\frac{d\hat{x}}{dt} = A\hat{x} + B\hat{u} + D\hat{d}$$

$$\hat{x} = \begin{bmatrix} T_{fn} & T_{fp} & P_s & V_w & V_b \end{bmatrix}^T$$

7.1)

$$\hat{u} = \begin{bmatrix} \dot{m}_{fuel} & \dot{m}_{fw} \end{bmatrix}^T$$

$$\hat{d} = \begin{bmatrix} \dot{m}_s & T_{fuel} & T_{fw} \end{bmatrix}^T$$

and the output matrix equation is given by:

$$\hat{y} = C\hat{x}$$

$$\hat{y} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{A_w} & \frac{1}{A_w} \end{bmatrix} \hat{x}$$

The linear model is calculated by the MATLAB function *linsim.*

8. Verification of the linear model

Verification of the linear model is also carried out at 50 percent load with pressure at 7 bars. The results are seen in figure 8.1, 8.2 and 8.3. The results shows that the dynamics behavior is captured well and a slightly difference in the static behavior. In spite of the slightly difference of the static behavior the linear model is suitable for model-based control.

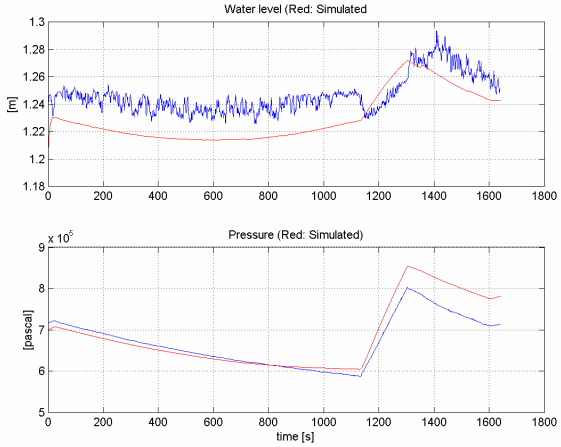


Figure 8.1: Step response on fuel flow

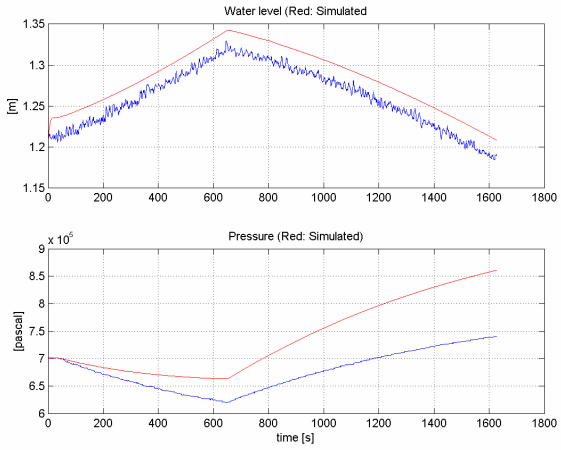


Figure 8.2: Step response on feed water flow

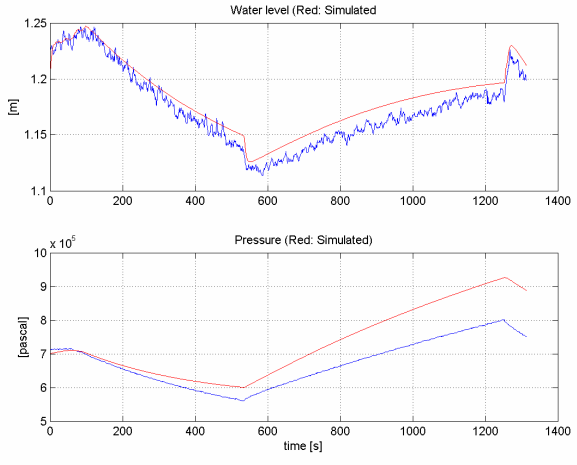


Figure 8.3: Step response on steam flow

9. Conclusion

A nonlinear physical model that is suitable for a model-based control has been presented. The model is based on physical parameters for the plant and can easily be scaled to represent any One Pass Smoke Tube Boiler. The nonlinear model agrees well with experimental data. In particular the shrink and swell phenomena are well captured by the model. The novelty in the model is the empirical equation. In order to be able to implement a linear controller the model was linearized and the linear model proved to inherit the properties of the nonlinear model. In the future the linear model will be utilized for designing a multivariable controller and test with the new controller design will be carried out on the test plant.

References

[1] *Homepage of "Aalborg Industries A/S"*,http://www.aalborg-industries.com/

[2] *Basic Design Manual - Aalborg Industries A/S*, proprietary.

[3] Lauritsen, Aage Birkekjær Gundtoft; Søren & Eriksen, Aage Bredahl *Termodynamik -Teoretisk grundlag - Praktisk anvendelse*, 1. edition, 2000 Ingeniøren Bøger (in danish).

[4] Dukelow, Sam G. *The Control of Boilers*, 2. edition, Instrument Society of America, 1991.

[5] *Homepage of MathWorks – MATLAB,* http://www.mathworks.com/products/matlab/

[6] Knudsen, Morten *Estimation of Physical Parameters in Linear and Nonlinear Dynamic, Systems*, Aalborg University, Department of Control Engineering, 1993.

[7] Nørregård Hansen, Niels; Uglsøe Hvistendahl, Poul; Just Nielsen, Rene & Solberg, Brian *Modelling and Control of a Marine Boiler*, Department of Control Engineering, Aalborg University, Institute of Electronic Systems, 2003.

[8] Sørensen, Kim, Houbak, Niels & Condra, Thomas *Modelling, Simulating and Optimizing Boilers*, In: 'Proceedings ECOS 2003', ECOS 2003, Technical University of Denmark, Denmark.

[9] Sørensen, Kim, Houbak, Niels & Condra, Thomas *Modelling of Boiler Heating Surfaces and Evaporator Circuits*, In: 'Proceedings SIMS 2002', SIMS-Scandinavian Simulation Society, 43nd SIMS Conference, University of Oulu , Finland.

[10] Sørensen, Kim; Karstensen, Claus M. S.; Houbak, Niels & Condra, Thomas *Modelling and simulating fire tube boiler performance*, In: 'Proceedings SIMS 2003', SIMS-Scandinavian Simulation Society, 44th SIMS Conference, Västerås , Sweden.

[11] Thorsen, R. & Landis, F. *Friction and Heat Transfer Characteristics in Turbulent Swirl Flow Subjected to Large Transverse Temperature Gradients*, Journal of Heat Transfer, p 87-96, February, 1968.

[12] Wagner W. & Kruse, A. *The industrial Standard IAPWS-IF97 for the Thermodynamic Properties and Supplementary equations for Other Properties - Properties of Water and Steam*, Springer Verlag, 1998

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

FLUID FLOW AND PARTICLE DEPOSITION SIMULATIONS IN THE HUMAN NOSE

Marit Kleven* and Morten C. Melaaen
Telemark University College (HiT-TF) and Telemark R&D Centre (Tel-Tek)
Porsgrunn
Norway

Martin Reimers[1,2], Jan Sigurd Røtnes[2,3], Lars Aurdal[3,4] and Per G. Djupesland[5]
University of Oslo[1], SimSurgery AS[2], Interventional Centre at Rikshospitalet[3], NR[4], and OptiNose AS[5]
Oslo
Norway

Abstract

In this work, the aerodynamics and particle flow patterns in the human nose are simulated by Computational Fluid Dynamics (CFD). The simulations were carried out by use of Fluent, and the volume grid was generated by Tgrid. To be able to perform successful CFD-calculations of the nose, construction of a proper surface grid of the nasal cavity was mandatory. A high resolution CT-scan was the first part of this process, whereafter the 3-dimensional surface geometry was created by a pipeline of image segmentation and- geometric modification steps. The work described in this paper was initiated by a Norwegian company (OptiNose AS) developing a patented concept for efficient nasal delivery of drugs and vaccines. The objective of the article is to show how CFD can be applied as a tool to visualize and demonstrate the basic features of this technique and how it can be used to further optimize design and function of novel delivery devices based on the bi-directional concept. It is worth of notice, that the complexity of the nasal airway represents considerable challenges for CFD computations. This article shows, however, promising results on the issue. The simulations in Fluent are carried out with a reasonable range of flow rates and particle sizes typical to nasal delivery.
Keywords: Computational Fluid Dynamics (CFD), Fluent, nose.

* Corresponding author. Phone: +47 3557 5190, Fax: +47 3557 5001, E-mail: marit.kleven@hit.no

Nomenclature

A	Projected area of particle [m^2]
C_D	Drag coefficient
F_i	Source term [$kg/(m^2 \cdot s^2)$]
g	Gravity force [m/s^2]
m_p	Particle mass [kg]
p	Pressure [Pa]
t	Time [s]
u	Velocity of the gas phase [m/s]
u_p	Particle velocity [m/s]
V_p	Volume of particle [m^3]
x_p	Particle position [m]
μ	Viscosity of the gas phase [$kg/(m \cdot s)$]
ρ	Density of the gas phase [kg/m^3]
ρ_p	Particle density [kg/m^3]
τ_{ij}	The stress tensor [Pa]

Introduction

The apparent external nose surrounds the nostrils and one-third of the nasal cavity, which according to Mygind [1] in its entirety consists of a 5 cm high and 10 cm long chamber. The nose consists of two nasal passages, which are separated by the nasal septum. The internal walls opposite the nasal septum have ridges formed of bone, i.e. conchae or turbinate bones. Several bones, the frontal, sphenoid, ethmoid and maxilla, contain hollow air-filled spaces, i.e. the sinuses. The olfactory mucosa

is situated in the roof of the nasal cavity, with correspondence to the brain. Approximately 1.5 cm from the nostrils is the narrowest and most resistive portion of the entire airway, i.e. the nasal valve. After passing through this narrow region, the air stream enters the nasal cavity, where the cross sectional area is much greater. According to Proctor [2], the direction of the air stream undergoes a bend of nearly 90° at this juncture. The bilateral airstreams increase in velocity as they join in the nasopharynx, which is the upper part of the throat, where another change in direction occurs.

The narrow and bent nasal geometry is essential for the important physiological functions of the nose, e.g. the ability to act as a filter and air-conditioner protecting the lower airways from harmful exposure. Nasal delivery is considered for an increasing number of existing and new drugs and vaccines. The narrow flow passages offer many challenges for efficient nasal delivery of drugs and vaccines. Current nasal delivery devices have major disadvantages and there is a considerable potential for improvement.

The flow of particles and droplets in fluids has a wide application in industrial processes, and CFD models are often used to study such problems. The objective for this work is to increase the knowledge of the particle-gas flow inside the human nose. Since CFD is a very useful tool when analyzing process equipment, it seems reasonable also to use this tool in the study of gas/particle flow. The complexity of the nasal airway and its physiological roles represent considerable challenges for CFD computations. The initiative to increase the knowledge of nasal aerodynamics came from the Norwegian company OptiNose AS, developing novel nasal delivery devices based on their patented delivery concept. The novel bi-directional nasal delivery concept takes advantage of the posterior connection between the nasal passages persisting when the soft palate automatically closes during oral exhalation, closing of the nasopharynx from its communication with the main pharyngeal airway. Exhalation into the delivery device triggers release of liquid or powder particles into an airflow, which enters one nostril via a sealing nozzle and exits through the other nostril. The aim of this article is to show how CFD can be applied as a tool to visualize and demonstrate the basic features of this technique and how it can be used to further optimize design and function of bi-directional delivery devices. CFD computations can increase the efficiency of device development and reduce the need for expensive and time consuming laboratory experiments. Thorough validation against physical experiments is of course essential for the reliability and value of such CFD computations.

Surface mesh generation

To be able to perform successful CFD-calculations of the nose, construction of a proper surface grid of the nasal cavity was mandatory. The first step of the modelling process was to acquire a high resolution CT-scan (The National Hospital, Norway). One of the authors was imaged in a CT scanner, generating 109 slices of the nasal geometry with 1 mm between each slice. Prior to further processing, these data were segmented, that is, each voxel in the data was given a unique label indicating whether it belongs to the airway or not. This is a classical problem in image processing, and many possible methods exist. We used a Markov Random field based method as these produce particularly smooth interfaces between the different groups of voxels, see Derin and Elliott [3].

The next step in the process was to produce geometry from the segmented image data, a surface triangular mesh that modelled the surface of the nasal cavity. The mesh was later used as input to the volume grid generator and was to comply with a set of constraints and quality measures dictated by the grid generator and the CFD software. These measures can be divided in two categories; geometric quality and mesh quality. We wanted on the one hand a mesh that closely modelled the geometry of the surface of the nasal cavity and on the other hand a mesh configuration that gave high quality volume grids of the cavity itself. Our overall strategy was to first generate an initial mesh directly from the segmented image. This mesh was typically unsuitable for the grid generation step, with far too many triangles, topological and geometric artefacts and suboptimal mesh structure. We therefore took a number of steps such as smoothing, decimation, refinement and mesh optimization to enhance its quality. Due to the often competing requirements, the nature of the enhancement process was iterative.

Initial mesh generation

Our first step was to employ the Marching Cubes (MC) algorithm of Lorensen and Cline [4], which in this setting can be seen as to extract the surface of the segmented voxels in a consistent and efficient manner. The resulting mesh revealed artefacts such as topological holes stemming from inaccuracies in the segmented data. One of the requirements to the mesh was that it had the correct topology and was a closed surface without singularities and self intersections. We found it convenient in this situation to go back to the 3D image and remove the artefacts in the images before generating an updated mesh with the MC algorithm. After that, the mesh was pruned for irrelevant parts and holes where filled by manually editing the mesh. Figure 1 illustrates the initial mesh generated with the marching cubes algorithm. Here, we can see the placement of the device in the left nostril.

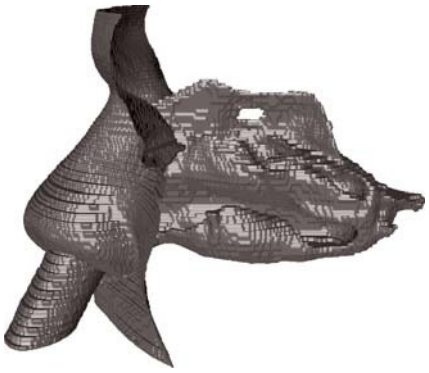


Figure 1: An initial mesh generated with the marching cubes algorithm.

Having successfully obtained a surface mesh with the right topology we could begin enhancing the geometry of the model.

Smoothing

The next step was to smooth the initial mesh, getting rid of the jagged geometry visible in Figure 1. One important requirement during the surface mesh production was that the mesh should be geometrically smooth whenever the underlying model was smooth. We used variants of Laplacian Smoothing, see Taubin [5], as it is a straightforward and common smoothing method. It is based on mesh topology alone and thus does not respect the local geometry of the mesh very well, especially where neighbouring triangles differ significantly in size. A more sophisticated method is presented in Desbrun et al. [6] where the intrinsic geometry of the surface is taken into account, giving a better geometric smoothing. We also found a rather effective smoothing method that we called “barycentre” smoothing as it amounts to moving each vertex to the average of the positions of its mesh neighbours. The first two methods where typically iterated until we got a visually pleasing mesh. The barycentre method on the other hand is very effective with one or two steps sufficient. A side effect that we learned to appreciate was that it also improved triangle shape significantly with many triangles being smoothed to be nearly equilateral. All of the smoothing methods mentioned had the tendency to shrink the volume enclosed by the surface. We countered this problem with a common workaround: scaling the vertex positions such that the volume remained the same after each smoothing step.

Decimation

The next step was to reduce the number of triangles down to a size that was manageable for the grid generator, typically 100 000 triangles. Mesh decimation algorithms such as the one in Garland and Heckbert [7] are typically based on a primitive mesh decimation operator such as the half edge collapse; a vertex is removed by identifying it with a neighbouring vertex, reducing the number of triangles with at least two. In addition a norm measuring the quality of the mesh is used. Based on this, a standard “greedy” strategy can be employed, where one in each step perform the decimation operation that reduce the geometric quality the least. In other words, we reduce the size of the mesh while keeping the most significant geometric information. In addition a set of constraints can be built in, such that e.g. operations that produce badly shaped triangles are not allowed. In principle we could have used the criterions listed in the section above to guide the decimation process, but we found that it was not necessary to maintain high mesh quality during the decimation process. We thus employed the algorithm described in [7] where the most important optimization criterion is geometric approximation quality. The other quality criterions were taken into account at a later stage. We reduced the number of triangles from more than a million to around 75 000 which amounted to 75% of our triangle budget. This gave us good enough approximations while leaving enough room for later refinement if necessary.

Optimizing triangle shapes

The next step was to improve the triangle shapes and distribution. One could think of this as the problem of redistributing the triangles so that they obtained better shape, while still representing roughly the same geometry. The input mesh to the grid generator must have nicely shaped triangles; ideally all triangles should be equilateral. "Skewness" measures how far a triangle is from being equilateral. An equilateral triangle has skewness 0 while a degenerate triangle has skewness 1. Our objective was to minimize the overall skewness and avoid triangles with skewness above 0.75.

The methods we used to optimize over skewness fall into two categories, optimizing over either vertex positions or vertex connectivity. We found the Laplacian Smoothing type of techniques described above particularly effective. The reason for the effectiveness was that these methods are based on averaging the geometry and thus the resulting mesh stays close to the geometry of the initial mesh. One step of barycentre smoothing was very effective in that respect. The other type of method we used was based on keeping the vertices of the mesh fixed while optimizing the vertex connectivity. The basic operation in the former type of method is the edge flip seen in Figure 2.

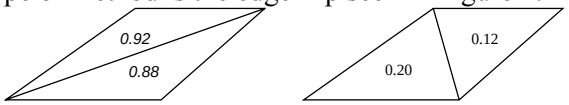


Figure 2: Edge flips improve the overall skewness of the triangle mesh.

We used the local optimization (LOP) technique [8, 9] with the cost for a triangle amounted to an exponential function of its skewness, on the form $c_i = Cs^p$, with C=10 and p=5. This ensured that triangles with high skewness were punished hard. The cost function in the LOP algorithm was the sum of this expression over all triangles. We also tested more advanced optimization techniques based on simulated annealing but concluded that the simple LOP algorithm was equally effective and produced meshes with few skew triangles. If there where still triangles with high skewness we improved their quality manually.

Refinement of "thin" areas

The grid generation process produces volumetric grids with element size proportional to the size of the surface triangles. In fact, a surface triangle in the input mesh ends up as a facet of a tetrahedron in the volumetric mesh. Therefore it is of vital importance that the density of the surface triangles is adapted to the thickness of the geometry. In other words, the size of the surface triangles is required to be proportional to the thickness of the geometry, with small triangles in thin regions. Thickness of the geometry could be defined in different ways. We define the thickness in a point on the surface to be the interior distance to other parts of the surface. In practice we estimated the local thickness at a triangle by measuring the distance to the nearest triangle in the interior normal direction. Due to the instable nature of these calculations, we employed a median filter to obtain consistent thickness values throughout the mesh. We found the results of this method satisfactory and sufficient for our purposes. Defining also the size of a triangle to be the length of its longest edge, we required the local thickness to be a factor 5 of the triangle size. We allowed in other words the volume to be at approximately five elements thick. In practice we adapted the local triangle density by adaptively refining the mesh in thin areas, splitting edges in two and dividing the triangles in smaller triangles. We repeated the following steps until our thickness criterion was met:

- Associate thickness with each triangle.
- Split triangles with too high triangle size to thickness-ratio.
- Optimize the split triangles with respect to triangle shape.

Partitioning the final surface mesh

After the above steps we would have a final surface mesh that met all our requirements with respect to geometric- and surface mesh quality. The final step was to divide the mesh into logical partitions each consisting of a set of connected triangles that would model inlet/outlet or other areas of special interest. We assigned each triangle to one unique partition and used this as part of the mesh description. The nature of the inlet/outlet partitions allowed us to use a simple region growing algorithm based on surface normals. We subsequently edited the partitions manually to

obtain visually smooth partitions. The final surface mesh is shown in figure 3.

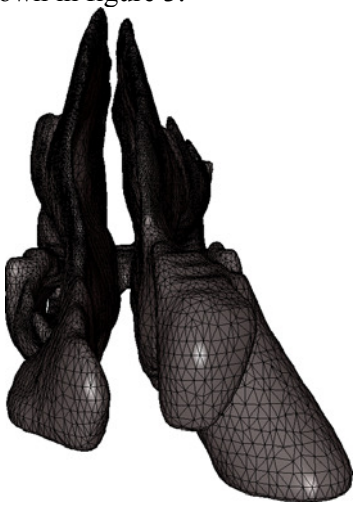


Figure 3: The final mesh modelling the surface nasal cavity.

Volume mesh generation

Figure 3 above shows the final mesh modelling the surface nasal cavity. By use of the default algorithm included in Tgrid, i.e. the highly automated Delaunay method [10], tetrahedrons in the nasal cavity volume were generated. The cell quality was tested in Tgrid, to ensure a proper grid for the simulations. This was done by investigating the degree of skewness, this time based on comparison of the cell's shape to an equilateral cell. Again, the skewness range is between zero and one, where zero skewness is optimal and a skewness of one indicates a degenerate cell.

Simulations

The mesh developed throughout the described process was imported into the CFD code Fluent. This code provides comprehensive modelling capabilities for a wide range of fluid flow problems, where a useful group of models is the multiphase flow models. According to Fluent [11], there are currently two approaches to the numerical calculation of multiphase flows; Euler-Lagrange approach and Euler-Euler approach. In this work, i.e. an analysis of a gas-particle flow, the discrete phase model (DPM) is used. This model follows the Euler-Lagrange approach. The gas phase is treated as a continuum by solving the Navier-Stokes equations, i.e. the conservation equations. In addition, the DPM performs Lagrangian trajectory calculations for the dispersed phase, i.e. the particles, including coupling with the continuous phase. In modelling, particles include both solid particles and droplets.

The fate of a particle in the air passage will be influenced by the dimensions of the space through which it passes and the nature of the airflow in that passage. The characteristics of air flow of importance are the velocity of the stream, bends in the passage, and the degree of turbulence of flow when the flow is turbulent. Other factors that influence the deposition and absorption of the medicaments from a nasal spray include particle size, particle size distribution, shape and density [2, 12]. The angle of the spray nozzle piece in the nasal vestibule and particle velocity from the nozzle also contributes. The latter was considered in this work, but will not be addressed in this paper.

The simulations were carried out with reasonable values for high and low air flows and different particle sizes. The distribution of particle sizes in a nasal spray was not considered. It was assumed that the density of the particles is equal to that of water. This is reasonable, because most nasal medications are diluted with water. It is furthermore assumed that the particles are spherical, which for droplets is a quite good assumption. Forces considered for the particle movement are drag, gravity and inertia. Brownian movement can also be involved in the likelihood of deposition, but only in relation to very small particles, and thus not considered in this work. The effect of gravity will, in proportion to the particles mass, influence its tendency toward sedimentation downward. Inertia forces will affect the fate of a particle in proportion both to its mass and the velocity of the stream in which it is suspended. This force is of special importance in relation to bends in the air stream caused either by turbulent flow or by a change in direction of the air passage. The only forces included for the particle movement in the final model are the inertia and drag force. The gravity force will depend on the position of the head, and since it usually is not well-defined how a person is positioned when using the nasal spray, this force was neglected. However, the gravity is easy to include in the model. While testing the model, it was seen that the gravity force only was important for large particles.

The flow in the nose is, according to Cole [12], neither turbulent nor laminar, but lies in the

transitional regime. Modelling a flow in the transitional regime is seen as a difficult task. Since the turbulence affects the distribution and uptake of medications in the nasal cavity in a positive manner, it was decided to simulate the airflow as a laminar flow in this study. This would rather under-estimate the degree of deposition than the opposite, if deviating from the real deposition pattern. Beyond, the conservation equations for mass and momentum for laminar flow are presented.

Equations

The equation for conservation of mass for the continuous phase, can be written as

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_i}\left(\rho u_i\right) = 0 \tag{1}$$

The right-hand-side of (1), i.e. the source term, is set to zero, since there is assumed no mass added to the continuous phase from the dispersed second phase, e.g. due to vaporization of liquid droplets. Conservation of momentum in the *i*th direction in a non-accelerating reference frame is described by

$$\frac{\partial\left(\rho u_i\right)}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho u_i u_j\right) = -\frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j} + \rho g_i + F_i \tag{2}$$

The last term on the right-hand-side of (2), F_i, represents the contributions from the dispersed phase. The stress tensor τ_{ij} is given by

$$\tau_{ij} = \left[\mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\right] - \frac{2}{3}\mu\frac{\partial u_l}{\partial x_l}\delta_{ij} \tag{3}$$

The gas phase is treated as a continuum by use of the above equations, while the dispersed phase is solved by performing trajectory calculations. In this work, it is assumed steady state for the continuum, therefore, the transient terms in equations (1) and (2) can be neglected.

The equation describing the velocity of an individual particle can be found by executing a force balance for one particle (vector equation);

$$\frac{d\left(m_p \vec{u}_p\right)}{dt} = \frac{1}{2}\rho C_D A\left|\vec{u} - \vec{u}_p\right|\left(\vec{u} - \vec{u}_p\right) + \vec{g}V_p\left(\rho_p - \rho\right) \tag{4}$$

The first term on the right-hand-side of equation (4) represents the drag force, and the next terms represent the gravity force. In addition we could have added other forces, like the lift force, virtual mass force and Basset force. During the simulations, only the drag force is included in the model. The other forces are easy to include, but not of importance in this study. Integrating equation (4) with respect to time yields the particle velocity. The time is the time from particle release. The position of the particle is calculated from

$$\frac{d\vec{x}_p}{dt} = \vec{u}_p \tag{5}$$

Numerical solution procedure

Fluent allows the use of either a segregated or coupled solver as the numerical method. In both cases, the CFD code will solve the governing equations for the conservation of mass and momentum. First, one divides the domain into discrete volumes using a computational grid. See Figure 3 for illustration of the surface mesh. The governing equations, described in the above section, are then integrated to produce a system of algebraic equations for the dependent variables in every computational cell, such as velocities and pressure.

In this work, the numerical method is the segregated solver. By this approach, the equations for the conservation of mass and momentum are solved sequentially, i.e. segregated from one another. Several iterations of the solution loop, is then required to meet a converged solution. The fluid properties are first updated by an initialized solution, before the momentum equations are solved using current values for pressure and face mass fluxes, in order to update the velocity field. A pressure correction equation is then solved to obtain the necessary corrections to the pressure and velocity fields and the face mass fluxes such that continuity is satisfied. When inter phase coupling is to be included, the source term, F_i, in the appropriate continuous phase momentum equation is updated with the discrete phase trajectory calculations. For relatively small particle size and low mass flow, the coupling between the two phases is weak. Then, we can exclude the inter phase coupling, in order to increase the speed of the calculation. Finally, a check for convergence of the equation set is made, and iterations stop when the convergence criteria is fulfilled. Otherwise, the fluid properties are updated, based on the current solution, and the steps are continued until the

convergence criteria is met.

Results

A typical result from the simulations of bi-directional flow pattern in Fluent is shown in Figure 4, where the nose geometry is included and a number of particles are released on the face of the left nostril medication unit. The figure shows the path of particles from the introduction into the left nostril and to the back of the nasal cavity, where the particles turn and follow the air stream out of the right nostril. The particle traces are coloured by the particle residence time. The colour scale in all the figures illustrates that the simulation results in this paper goes from red representing high values, to blue representing low values.

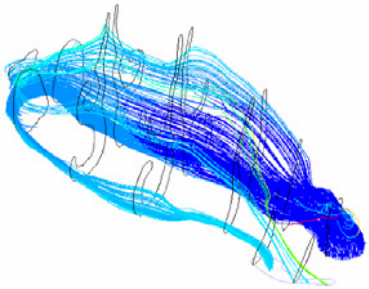


Figure 4: Flow of particles in the nasal cavity with low air velocity and small particle size. The darker lines show the cross-sections.

We can see from Figure 4 that the number of particles in the left nostril is considerably higher than in the right nostril. The difference in particles entering and exiting the nostrils is assumed to be the amount of particles that deposit in the nasal cavity. The same flow of particles seen from above is illustrated in Figure 5.

Figure 5: Flow of particles with low air velocity and small particle size seen from above.

We can also focus on the cross-sections of the nasal cavity, illustrated by the darker lines in Figure 4, and study the gas velocity distribution.

Figure 6: The velocity distribution in the cross-sections of the nasal cavity with a low air velocity.

In the simulations resulting in Figures 4 and 5 the air flow is low. When increasing the air velocity, and simulating with the same size of the particles, less particles follow the air stream around the nasal septum and out the right nostril. Because of inertia the particles with higher velocity will not manage to follow the air flow and the particles hit the surface and deposition appears. Figure 7 illustrates this situation.

Figure 7: Flow of particles in the nasal cavity with high air velocity and small particle size.

As mentioned, the particle size greatly influences the deposition of particles from the nasal spray. Figure 8 is showing an equal number of larger particles being released on the face of the left nostril medication unit, compared to the particles being released in Figure 4. The air velocities in the two figures are identical. Figure 8 shows that when increasing the particle size less particles follow the air stream around the nasal septum and out the right nostril. This is due to inertia forces.

Figure 8: Flow of particles in the nasal cavity with low air velocity and large particle size.

Comparison of the simulations with initial gamma-scientigraphic studies is promising.

Conclusions and further work

The effects of increased velocity and increased particle size are seen. This work shows that by starting with a CT-scan in hospital, one can establish a nasal geometry surface mesh, a volume mesh, and then run simulations on the nasal cavity. It is thus possible, that CFD computations can increase the efficiency of device development and reduce the need for expensive and time consuming laboratory experiments. Thorough validation against physical experiments is of course essential for the reliability and value of such CFD computations.

Analyzing the flow pattern more thoroughly, e.g. evaluating the need for testing different turbulence models included in Fluent, will be done in the future. Perhaps the use of one of the models will improve the results further.

Since a nasal spray has a typical particle size distribution, it might be interesting to run simulations with such a distribution in the future, instead of equally sized particles.

References

[1] Mygind N., Dahl R. *Anatomy, physiology and function of the nasal cavities in health and disease.* Advanced Drug Delivery Reviews 29, 3-12, Elsevier Science B.V., 1998.

[2] Proctor D.F., Andersen I. *The nose – Upper airway physiology and the athmospheric environment.*, 1982.

[3] Derin H., Elliott H. *IEEE Transactions on Pattern Analysis and Machine Intelligence, Modeling and Segmentation of Noisy and Textured Images Using Gibbs Random Fields*, no. 1, vol. 9, pp. 39-55, January 1987.

[4] Lorensen W.E., Cline H.E. *Marching Cubes: a high resolution 3D surface reconstruction algorithm*. In SIGGRAPH 87 Proceedings.

[5] Taubin G. *A Signal Processing Approach to Fair Surface Design*, Computer Graphics 29, pp. 351-358, 1995.

[6] Desbrun M., Meyer M., Schröder P., Barr A.H. *Implicit Fairing of Arbitrary Meshes using Diffusion and Curvature Flow*. In ACM Siggraph '99 Proceedings.

[7] Garland M., Heckbert P.S. *Surface Simplification Using Quadric Error Metric.* In SIGGRAPH 97 Proceedings.

[8] Dyn N., Hormann K., Kim S.J., Levin D. *Optimizing 3d triangulations using discrete curvature analysis*. In mathematical methods for curves and surfaces: Oslo 2000, Lyche T., Schumaker L.L. (eds.), Vanderbilt University Press, pp. 135-146, 2001.

[9] Lawson, C. *Transforming Triangulations. Discrete Mathematics 3*, pp. 365-372, 1972.

[10] *Tgrid 3 User's Guide, First Edition*. Fluent Inc., Centerra Resource Park, Lebanon, June 1997.

[11] *Fluent User's Guide,Release 6.1*. Fluent Inc., Centerra Resource Park, Lebanon, January 2003.

[12] Cole P. *The respiratory role of the upper airways - A selective clinical and pathophysiological review*. St. Louis, United States of America: Mosby-Year Book, Inc., 1993.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

TOWARDS AUTOMATIC GENERATION OF MODEL CHECKABLE CODE FROM MODELICA

Håkan Lundvall, Peter Bunus and Peter Fritzson
Department of Computer and Information Science,
Linköping University, Sweden

Abstract

Using model components in complex system modeling is sometimes difficult because many semantic properties that should be obeyed during the design are not formalized in the modeling language. There exist rules that users of the components should follow in order to create semantically, mathematically, and physically correct models. Program verification aims at proving that programs meet certain specifications, i.e. that the actual program behavior fulfils certain specified properties. Model checking is a specific approach to verification of temporal properties of reactive and concurrent systems. Verification is usually carried out by using model checking algorithms to demonstrate the satisfiability of certain properties formalized as logical formulae over the model of the system. The model checking approach has proven successful for models based on finite-state automata and is based on state space inspection. To realize the full potential of the simulated and modeled systems with Modelica it is important to verify important properties of models in order to ensure that they meet the required criteria. In this paper we describe an algorithm that translates a non-trivial subset of Modelica to the model checking language of HyTech.

Keywords: Modelica, Model checking

Nomenclature

In the examples the following variables and symbols are used:

alpha	Time delay parameter in the railroad example
app,exit	Events indicating approach and exit of the train.
clk1, clk2	Discrete variables storing time-stamps.
delay	Stopwatch variable with time derivative 0 or 1.
dy	Time derivative of y.
lower	Boolean value that when becoming true tells the gate to close.
open	Boolean value representing the state of the valve.
raise	Boolean value that when becoming true tells the gate to open.
t	Time variable.
v	Velocity of the train.
s1, …, s4	States of the hybrid automata.
x	Distance from the train to the gate.
y	Water level.

Introduction

Modelica (Fritzson 2003 [1]) is an object oriented modeling language capable of describing heterogeneous physical system. To take full advantage of the modeling and simulation capabilities of Modelica it is desirable to be able to formally verify important properties of the model. In order to do that the discrete algebraic equations that the Modelica model results in must be translated into a form that a tool capable of performing automated program verification can use. The hybrid automata formalism is such a form where tools like HyTech (Henzinger et al. 1997 [2]) and CheckMate (Kapinski and Krogh 2002 [3]) can perform program verification.

Model Checking

Model checking (Clarke et al. 1999 [4]) is a technique to perform program verification over finite state concurrent systems in a fully automated way. It is done by exploring the state space of the system. Validation is carried out against a formal specification consisting of a set of formulas over the model of the system. These formulas usually express safety properties that should be fulfilled at all times or illegal states that the system should be proven never to enter.

The two main advantages compared to other program verification techniques is that it can be fully automated and that it produces a sequence of steps in the specification leading to the state that does not satisfy the specification. This makes model checking suitable for integration in automated debugging and verification tools.

The specification consists of a temporal logic formula expressing the desired properties of the system. The notion of the Kripke structure is used to model the behavior of the system. A Kripke structure consists of a set of states and transitions between states. Each state also has a valuation for a set of properties that are either true or false in the state. A transition from one state to another represents a step in time that makes a distinguishable difference of the state of the system. A path in a kripke structure represents a computation of the system.

Many systems lead to very large state spaces. In order to be able to handle this, symbolic algorithms can be used. This is especially important in model checking of hybrid systems since real valued variables are present and the value ranges of these variables need to be represented symbolically.

A temporal logic often used in model checking is called Computational Tree Logic (CTL). CTL formulas describe properties of computation trees. A computation tree is formed by selecting one of the states in the kripke structure to be the initial state. The rest of the infinite tree is then formed by unwinding the kripke structure from the initial state.

CTL formulas are composed of path quantifiers and temporal operators. Path quantifiers express properties of the branching structure of the tree. The two path quantifiers in CTL are **A** and **E**, representing “for all paths” and “for some path”, respectively. These are used at a state to express that some property holds for all or some of the paths starting at that state. The temporal operators express properties of a path through the tree. The five basic operators are presented in Table 1.

Symb.	Phrase	Explanation
X	Next time	Requires a property to hold in the second state of the path.
F	Eventually	Requires a property to hold at some state along the path.
G	Always	Requires a property to hold at every state of the path.
U	Until	This operator combines two properties and requires that there is a state on the path where the second property holds and that at every prior to that state, the first property holds.
R	Release	This operator also combines two properties and requires the second property to hold at every state up to and including the first state at which the first property holds.

Table 1. Basic operators of CTL.

CTL formulas are divided in two categories; state formulas and path formulas. State formulas hold at a specific state in the Kripke structure. Path

formulas hold along a specific path in the kripke structure.

The syntax of a formula in CTL follows the following rules:
All atomic propositions are state formulas. If *f* and *g* are state formulas then $\neg f, f \vee g, f \wedge g$ are state formulas. If *f* is a path formula then **E** *f* and **A** *f* are state formulas. If *f* is a state formula then *f* is also a path formula. If *f* and *g* are path formulas then $\neg f$, $f \vee g$, $f \wedge g$, **X** *f*, **F** *f*, **G** *f*, *f* **U** *g*, *f* **R** *g* are path formulas.

An example of a CTL formula that holds if when a request occurs then it will eventually be acknowledged: **AG**(*Req* → **AF** *Ack*).

Hybrid Automata

Existing model checking tools for hybrid systems relies on the formalism of hybrid automata, which is an augmented form of finite automata where a finite set of continuous variables are allowed.

The hybrid automaton is formally defined as follows:

Variables: A finite set $X = \{x_1, x_2, ..., x_n\}$ of real-valued variables. E.g. y in the Figure 2.

Control modes: A finite set *V* of control modes. These correspond to states in a finite automaton.

Flow conditions: A labeling function *flow* that assigns a flow condition to each control mode $v \in V$ The flow condition *flow(v)* is a predicate over the variables in $X \cup \dot{X}$. While an automaton is in a control mode *v* the variables in *X* evolve along a curve such that at all point along the curve the values of the variables and their first derivative satisfy the flow condition. In Figure 2 the equations involving `dy` represent the flow conditions.

Invariant condition: A labeling function *inv* that assigns an invariant condition to each control mode *v*. The invariant condition *inv(v)* is a predicate over the variables in *X*. While a hybrid automaton is in a control mode *v*, the variables in *X* must satisfy the invariant condition *inv(v)*. Represented in Figure 2 by the inequalities inside the ellipses.

Initial conditions: A labeling function *init* that assigns initial conditions to each control mode v. The initial condition *init(v)* is a predicate over the variables in *X*. The control of a hybrid automaton may start in the control mode *v* when the initial condition *init(v)* is true.

Control switches: A finite multiset *E* of control switches. Each control switch is a directed edge between a source mode $v \in V$ and a target mode $v' \in V$. Control switches are denoted by arrows in Figure 2.

Jump conditions: A labeling function *jump* that assigns a jump condition to each control switch in *E*. The jump condition *jump(e)* is a predicate over the variables in $X \cup X'$. The symbols *X* refer to the values of the variables before the control switch and the symbols in *X´* refer to the values of the variables after the control switch. The label `delay'=0` in Figure 2 denote a jump conditions stating that the value of `delay` is zero after the control switch.

Events: A finite set $\sum$ of events and labeling function *syn* that assigns an event in $\sum$ to each control switch in *E*.

This definition was taken from [2].

In order to be able to perform automatic analysis on hybrid systems certain restrictions have to be put on them. One such restricted class of the hybrid automaton is the linear hybrid automaton. In a linear hybrid automaton the flow conditions are predicates over the derivatives only, so that the derivative of a variable cannot be a function of any variable in *X*. Further more, the flow conditions, the invariant conditions and the initial conditions are convex linear predicates and for every control switch the jump condition is a convex linear predicate.

HyTech

HyTech is an automatic tool for the analysis of embedded systems. We decided to use HyTech as a target platform for our translator since it allows the symbolic verification and algorithmic analysis of hybrid dynamic systems. However, HyTech can only model linear hybrid automata, which is limiting for most of the simulation models expressed in Modelica. In order to overcome this limitation, we intend to extend our translator to generate code for HyTech+ (Henzinger et al. 2000 [5]) and CheckMate. HyTech+ and CheckMate are both capable of verifying hybrid systems with general continuous expressed with linear and non-

linear differential equations. The input to the HyTech tool consists of a definition of the automaton to check and an analysis section where the specification is represented. It is possible to enter more then one automaton, in which case the HyTech tool transforms them into a single automaton as its initial step. In this step *synclabs*, synchronization labels in HyTech, can be used to synchronize control switches in different automata. Two control switches marked with the same synclab always triggers simultaneously.

Translator algorithm

Modelica is based on the mathematical formalism of DAE (Differential Algebraic Equations) while HyTech uses the linear hybrid automata formalism, therefore we must find a way to isolate the discrete control modes of the model in order to build an automaton that is equivalent to the original model. Since HyTech only handles linear hybrid automata only Modelica models with state variables of constant derivatives can be translated. In models where both constant and variable derivatives exist it is possible to use program slicing (Hatcliff et al. 2000 [6]) to isolate the linear part if there are no dependencies from the linear variables to nonlinear variables.

We start the translation by creating a startup mode and then create transitions to new modes for each Modelica when-equation that can be triggered from that mode. For each new mode the procedure is then repeated according to the stepwise description of the algorithm below.

The steps of the algorithm

Step one: Find all state variables (`y` in the water tank example below) and add them as continuous variables to the HyTech model. Also add the clock variable `t`. In Modelica there is a built in variable `time` present.

Step two: Find all discrete variables that are used together with continuous variables in when-equations (`open` in the water tank example). The discrete variables that do not appear together with continuous variables are not added, but they may appear as synclabs later.

Step three: Find the start value of all state variables and discrete variables and generate a starting control mode in the automaton. In the water tank example we have `open=true` and `y=1` as the initial condition.

Step four: Find all when-equations that can be triggered from this control mode. When-equations with conditions that are already true or cannot become true in the given mode are discarded; all other when-equations result in corresponding transitions in the hybrid automaton. In the water tank example we have `delay(y,2) >= 10 and open`. The other clause of the condition can never be triggered in this mode since `dy` is positive.

Step five: The new transitions result in new values for the variables and the derivatives. If there is an already added control mode that matches all the values, then make the transition point to that control mode; otherwise create a new mode in the automaton. For all added modes start over from step four.

When-equations that only depend on discrete variables trigger only as a result of another mode switch, since discrete variables only change at events. If the when-equation that changes the discrete value resides in the same component then the transitions are merged into a single transition. If the source event resides in a different component a synclab is generated so that the mode switches in the two resulting automata occur simultaneously. If there is no source event that can trigger the transition, then it is omitted.

When a transition that has a guard involving continuous variables is added to the automaton, a corresponding inequality has to be added to the invariant condition of the control mode, otherwise the automaton would have the possibility to stay in the source control mode and not follow the transition.

In Modelica there is a `delay` operator that delays a signal a specified amount of time. In order to translate this into the hybrid automaton an extra control mode is inserted in transitions involving a delayed variable. This extra control mode is a copy of the source control mode of the transition where the delayed transition is replaced by a transition depending on a clock variable. See the WaterTank example below for details.

Example 1: WaterTank

This example is taken from the HyTech system user's guide. Consider a water tank that is leaking water at a constant rate. When the water level falls

below five, a sensor signals a valve to open, which results in the tank being filled at a constant rate. However, the signal is delayed for two seconds before the valve reacts on it.

When the water level reaches ten the valve is signaled to close, again with a two second delay. The Modelica code for such a system is shown in Figure 1.

```
model WaterTank
  Boolean open(start=true);
  Real y(start=1);
equation
  when delay(y, 2) >= 10 and open
    or delay(y, 2) <= 5 and not open
  then
    open = if pre(open) then
             false else true;
  end when;

  der(y) = if open then 1 else -2;
end WaterTank;
```

Figure 1. WaterTank in Modelica.

The proposed algorithm would result in the automaton shown in Figure 2. In the figure each control mode is denoted by an ellipse and the arrows between them represent control switches. The slopes of the variables and the invariant regions are presented inside the ellipse. Each control switch is labeled by jump conditions, e.g., "`y ≥ 10`" and "`delay'=0`", which state that in order for that control switch to take place the value of `y` before the switch must be grater than 10 and the value of `delay` after the switch is 0.

Running HyTech on the resulting automaton shows us that the water level is kept between 1 and 12 at all times.

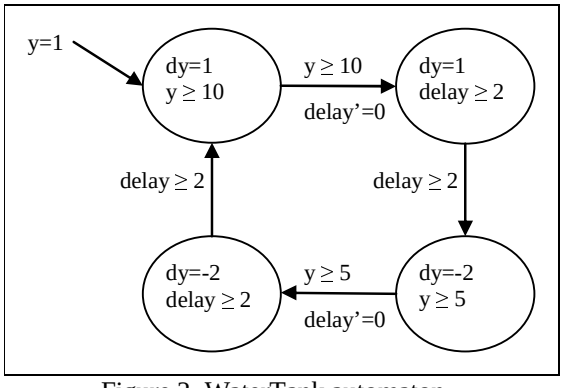


Figure 2. WaterTank automaton.

Example 2: Railroad crossing

This example, also taken from the HyTech user's guide demonstrates the ability to use model checking to calculate safe parameter values

The example consists of a train that passes a railroad crossing. At a distance of 1000 m a signal is sent to the controller to lower the gate and 100 m past the crossing a signal is sent to raise the gate. The parameter alpha in the controller is the delay from when the signal is sent until it is reacted upon. Here model checking is used to calculate safe values for alpha so that it can be guaranteed that the gate is closed whenever the train is closer than 10 m from the crossing.

The Modelica model is divided into three components that are shown in the Figure 3, the Figure 4 and the Figure 6.

```
import
 Modelica.Blocks.Interfaces.BooleanPort;

package Railroad
  model train
    BooleanPort app;
    BooleanPort exit;
    Real x;
    discrete Real v;
  initial equation
    v = -45;
    x = 2000;
  equation
    der(x) = v;
    when x >= 100 then
      exit = app;
    end when;
    when x <= 1000 then
      v = -40;
    elsewhen x <= 0 then
      v = 35;
    end when;
  end train;
```

Figure 3. Train model in Modelica.

If we first look at the train model, the only state variable is `x`, so `x` is added to the variables of the resulting automata. An initial control mode `s1` is also added to the train automata. Since the derivative of `x` is negative in the initial state only the event occurring when `x` goes below 1000 is added as a transition from this state. A new control mode, `s2`, is created as a target for the transition since there are no existing modes that match the variable values. In this control mode the only relevant event is when `x` passes zero. This

results in a new transition and a new control mode, and so on. The resulting automaton is shown in Figure 5.

```
  model gate
    Real y;
    discrete Real dy;
    BooleanPort lower;
    BooleanPort raise;
  initial equation
    dy = 0;
    y = 90;
  equation
    der(y) = dy;
    when lower then
      dy = -9;
    elsewhen raise then
      dy = 9;
    elsewhen y <= 0 then
      dy = 0;
    elsewhen y >= 90 then
      dy = 0;
    end when;
  end gate;
```

Figure 4. Gate model in Modelica.

The gate component results in one new continuous variable `y`. Since `y` is constant in the initial state of the gate automaton, there is no restriction in the invariant region of this state, but there are two events in the Modelica model that needs to be handled. Since the conditions of these events depend only on discrete variables we must search for the events that make them change. In this case there are events in the controller component that changes the variable. This results in the synclabs `lower` and `raise` that are added to both of the controller automaton and the gate automaton. The code for the gate automaton is shown in Figure 7.

```
var
  x, y : analog;
  t : clock;
  clk1, clk2 : discrete;
  alpha : parameter;
automaton train
synclabs: app, exit;
initially s1 & x = 2000;
loc s1: while x >= 1000 wait { dx=-45 }
  when x <= 1000 sync app goto s2;
loc s2: while x >= 0 wait { dx=-40 }
  when x <= 0 goto s3;
loc s3: while x <= 100 wait { dx=35 }
  when x >= 100 sync exit goto s4;
loc s4: while True wait { dx=35 }
end
```

Figure 5. Train automaton in HyTech.

In the controller automaton there exist no state variables but there are two discrete variables that are used in expressions together with continuous variables in when-equations; therefore they are added as discrete variables to the HyTech model. See Figure 8.

```
  model controller
    parameter Real alpha=1.0;
    BooleanPort lower;
    BooleanPort raise;
    BooleanPort app;
    BooleanPort exit;
    discrete Real clk1(start=0);
    discrete Real clk2(start=0);
  equation
    when app then
      clk1 = time;
    end when;
    when exit then
      clk2 = time;
    end when;
    when app and time-clk1 > alpha then
      lower = true;
    end when;
    when exit and time-clk2 > alpha then
      raise = true;
    end when;
  end controller;

  model test
    controller ctrl;
    train tr;
    gate g;
  equation
    connect(tr.app, ctrl.app);
    connect(tr.exit, ctrl.exit);
    connect(g.lower, ctrl.lower);
    connect(g.raise, ctrl.raise);
  end test;
end Railroad;
```

Figure 6. Controller model in Modelica.

```
automaton gate
synclabs: lower, raise;

initially s1 & y = 90;

loc s1: while True wait {dy=0}
  when True sync lower goto s2;
  when True sync raise goto s3;

loc s2: while y >= 0 wait {dy=-9}
  when y <= 0 goto s1;

loc s3: while y <= 90 wait {dy=9}
  when y >= 90 goto s1;
end
```

Figure 7. Gate automaton in HyTech.

```
automaton controller
synclabs: lower, raise, app, exit;

initially s1;

loc s1: while True wait {}
  when True sync app
       do {clk1'=t} goto s2;
  when True sync exit
       do {clk2'=t} goto s3;

loc s2: while t-clk1 <= alpha wait {}
  when t-clk1 >= alpha sync lower
       goto s1;
  when True sync exit
       do {clk2'=t} goto s3;

loc s3: while t-clk2 <= alpha wait {}
  when t-clk2 >= alpha sync raise
       goto s1;
  when True sync app
       do {clk1'=t} goto s2;

loc s4: while t-clk1 <= alpha &
              t-clk2 < alpha wait {}
  when t-clk1 >= alpha sync lower
       goto s3;
  when t-clk2 >= alpha sync raise
       goto s2;
end
```

Figure 8. Controller automaton in HyTech.

In order to make an analysis of the system we must add a section containing the analysis commands. These are shown in Figure 9.

```
var init_reg, reached, avoid: region;

avoid := y > 0 & x <= 10;

init_reg := loc[train] = s1 & x = 2000 &
            loc[gate] = s1 & y = 90 &
            loc[controller] = s1;

reached :=
   reach forward from init_reg endreach;

print omit all locations
      hide non_parameters in
        reached & avoid
      endhide;
```

Figure 9. Analysis commands for railroad example in HyTech.

The region `avoid` represents the forbidden condition. By printing the resulting region of the intersection between the reachable region and the forbidden region while hiding all non-parameter values and locations we arrive at an expression for the values of `alpha` that leads to unsafe states.

Related Work

Model checking can successfully complement existing software quality assurance techniques such as testing and debugging. Therefore it is important to provide efficient translators from various programming languages to model checkable formal languages. In this way, mature model checking techniques can be reused and applied to software systems that otherwise would not provide support for proving safety and liveness properties. Bridging the gap between high level languages such as C, C++, Java, Ada and Modelica and the input required by model checking tools (finite state automata with properties formulated in temporal logic) require the development of complex tool sets. In this section we present some of the translation frameworks that are most related to ours.

The SLAM project at Microsoft Research (Ball and Rajamani. 2002 [7], Ball and Rajamani 2001 [8]) checks temporal safety properties of sequential C programs. The system requires that the checked properties are encoded in a language called SLIC (Specification Language for Interface Checking).

The Bandera tool set (Corbett et al. 200 [9]) is an integrated collection of program analysis, transformation and visualizations components that enables the extraction of finite state models from Java source code. Bandera is able to generate a description of a finite-state transition system in the Promela and Trans languages that can be interpreted by the SIPN and SMV (Symbolic Model Verifier) models checking systems. Previously to the Bandera project, the same research group at Kansas State University has developed a toolset for translating Ada source code to the input language of the SPIN and SMV model checkers (Dwyer et al. 1998 [10]).

A related project to Bandera is the Java PathFinder (Brat et al. 2000 [11]) that translates Java programs to Promela, the specification language of the Spin model checker. Java PathFinder can detect race conditions, deadlocks, and violations of user specified assertions. The tool has been incorporated as a back-end checker for Bandera.

Conclusion

In this paper we have briefly outlined an algorithm to translate Modelica models to a representation that can be automatically verified against a formal specification using model checking. To be able to perform verifications on more sophisticated models it is possible to continue along this path and generate code for other systems such as CheckMate and HyTech+, both using hybrid automata.

The presented work in this paper should be seen as an important component for a broader attempt to make static analysis (Bunus and Fritzson 2004 [12]), run-time verification through algorithmic debugging (Bunus and Fritzson 2003 [13]) and model checking techniques, more applicable for the development of new automatic debugging tools with enhanced user-interaction for the Modelica language. We intend to implement a prototype translator using the presented algorithm in the numeric and symbolic engine developed for the OpenModelica compiler back-end.

References

[1] Fritzson Peter. (2003). *Principles of Object-Oriented Modeling and Simulation - with Modelica 2.1.* IEEE Press and John Willey, 2003.

[2] Henzinger Thomas, Pei-Hsin Ho, and Howard Wong-Toi. (1997) "HyTech: A Model Checker for Hybrid Systems." *International Journal on Software Tools for Technology Transfer*, vol. 1: 1-2, pp. 110-122, 1997.

[3] Kapinski James and Bruce H. Krogh. (2002). "A new tool for verifying computer controlled systems." In *Proceedings of the IEEE Conference on Computer-Aided Control System Design.* (Glasgow, Scotland, September, 2002).

[4] Clark Edmund M., Grumberg Orna, Peled Doron A (1999). Model Checking. MIT Press. 1999.

[5] Henzinger Thomas A., Ben Horowitz, Rupak Majumdar, and Howard Wong-Toi, "Beyond HyTech: Hybrid System Analysis Using Interval Numerical Methods," in *Hybrid Systems: Computation and Control. Lecture Notes in Computer Science 1790*, B. Krogh and N. Lynch, Eds.: Springer Verlag, 2000, pp. 130-144.

[6] John Hatcliff, Matthew B. Dwyer, Hongjun Zheng "Slicing Software for Model Construction". *Journal of Higher-Order and Symbolic Computation*, Volume 13, Issue 4, December 2000, Pages 315 - 353

[7] Ball Thomas and Sriram K. Rajamani. (2002). "The SLAM Project: Debugging System Software via Static Analysis." In *Proceedings of the 9th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages.* (Portland, Oregon USA, January 16-18, 2002).

[8] Ball Thomas and Sriram K. Rajamani. (2001). "Automatically Validating Temporal Safety Properties of Interfaces." In *Proceedings of the The 8th International SPIN Workshop on Model Checking of Software.* (Toronto, Canada, May 19-20, 2001). LNCS 2057, Springer Verlag.

[9] Corbett James, Matthew Dwyer, John Hatcliff, Corina Pasareanu, Robby, Shawn Laubach, and Hongjun Zheng. (200). "Bandera: Extracting Finite-state Models from Java Source Code." In *Proceedings of the 22nd International Conference on Software Engineering.* (Limerick, Ireland, June 4-11, 200).

[10] Dwyer Matthew B., Corina S. Pasareanu, and James C. Corbett (1998) "Translating Ada programs for model checking: A tutorial.," Kansas State University, Department of Computing and Information Sciences, Technical Report 98-12, 1998.

[11] Brat Guillaume, Klaus Havelund, Seung Joon Park, and Willem Visser. (2000). "Java PathFinder Second Generation of a Java Model Checker." In *Proceedings of the In Proceedings of Post-CAV Workshop on Advances in Verification.* (Chicago, July, 2000).

[12] Bunus Peter and Peter Fritzson. (2004) "Automated Static Analysis of Equation-Based Components." Simulation: Transactions of the Society for Modeling and Simulation International. Special Issue on Component Based Modeling and Simulation. To appear, 2004.

[13] Bunus Peter and Peter Fritzson. (2003). "Semi-automatic Fault Localization and Behaviour Verification for Physical System Simulation Models." In *Proceedings of the 18th IEEE International Conference on Automated Software Engineering.* (Montreal, Canada, October 6-10, 2003).

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

GRIDMODELICA - A MODELING AND SIMULATION FRAMEWORK FOR THE GRID

Kaj Nyström, Peter Aronsson and Peter Fritzson Linköping University
Department of Computer and Information Science
Sweden

Abstract

Simulation of complex models is a computationally expensive task. With the advent of grid computing, modelers can gain access to vast amounts of cheap computational power. This has however up until now required quite some effort to be put into specially written simulations in Fortran or C and also in the deployment of the simulation on the grid; a set of interconnected computers behaving as one single computational resource from an end-user perspective. We propose a framework called GridModelica for transparently creating and deploying simulations in the high level modeling language Modelica on computational grids. The first step in this framework is taken with the application GridParamSweep which demonstrates how easily parameter studies can automatically be performed in a grid environment.

Keywords: GridModelica, OpenModelica, Grid Computing, Modelica, parameter sweep

Introduction

One of the greatest problems with modeling and simulation today is the high computational costs associated with simulation of complex models. Many simulations require the computer to solve systems with many hundreds of thousands of equations, which can take quite some time even on powerful workstations. With the application GridParamSweep we take the first step towards putting the power of grid computing in the hands of modelers in order to minimize time- and cost consumption for simulating complex models.

Already today, lots of complex simulations are run on high performance computers and clusters and even on computational grids. To the best of our knowledge though, these models are almost without exception specially written in Fortran, C or similar languages. While these languages can provide high performance, if sufficiently well written, they do not really comply with the needs of today's modelers.

Instead we use the mathematical modeling language Modelica [1, 2] which combines the power of modern solvers written in Fortran or C with a high level of abstraction, object orientation and reuse. This, in combination with the computational power of the grid will provide a cheap high-performance platform for complex simulations.

To achieve the goal of providing the possibility of transparently running simulations on the grid we propose a framework called *GridModelica*. GridModelica will be both a language extension of the Modelica modeling language and a modeling and simulation toolkit, which enables modeling and simulation targeting the grid. As a first example we study how automated parameter studies can be performed in this framework. Such problems can be used for instance in parameter design optimization to find the optimal set of parameters for a given model. These examples are suitable for execution on the grid due to the nature of the problem with many independent simulations. These simulations can be executed on different computers with no communication between them, resulting in little overhead.

Modelica

In this paper the model intended for simulation is specified in the mathematical modeling language Modelica [1], which allows acausal modeling of heterogeneous systems spanning many different domains, such as for example electrical, mechanical and thermo fluid problems. Modelica is object oriented which allows for a high level of reuse and or-

dering of objects into hierarchies.
The modelica model can be either written by hand or specified using a graphical model editor as pictured in Figure 2. Either way, the result is an textual model which can be compiled into an executable which in turn can be run independently of the modeling environment, making it very suitable for grid deployment.

The Grid

A computational grid is an interconnected set of computers which together work as a single computational unit from an end-user perspective. A definition from the book "The Grid" [3] is "a hardware and software infrastructure that provides dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities". The word "grid" is borrowed from the electrical power grids, a much used analogy in grid computing. The idea is that when you plug in for example your toaster to an electrical outlet, you do not care where the electrical power you are using is produced. You just plug in an use it. A computational grid is meant to work in the same way. You do not care about where your job is executed. You just submit your job and then eventually fetch your results. The grid middleware takes care of the rest. A schematic sketch of the grid is pictured in Figure 1.
We would like to point out that using the grid is not yet as simple as using the electrical power grid. However a lot of development is still going on in the area of grid middleware and improvements are constantly being made. In the GridModelica project we use the Nordugrid middleware [4] which is closely related to the de facto standard the Globus toolkit [5]. The Nordugrid middleware has been deployed on both the Nordugrid testbed and the Swegrid [6] computational grid, both of which have been used successfully with the GridParamSweep application. The Nordugrid currently (2004-05-15 13:45 CET) encompasses 2327 processors on 35 different sites, providing from one to 644 processors each. The sites are distributed throughout ten different countries all over the world but concentrated to Scandinavia. Resources are shared with local users as the cluster managers sets the priorities which means that normally, only idle processors are used by the grid, thus exploiting a resource which would otherwise have been wasted.

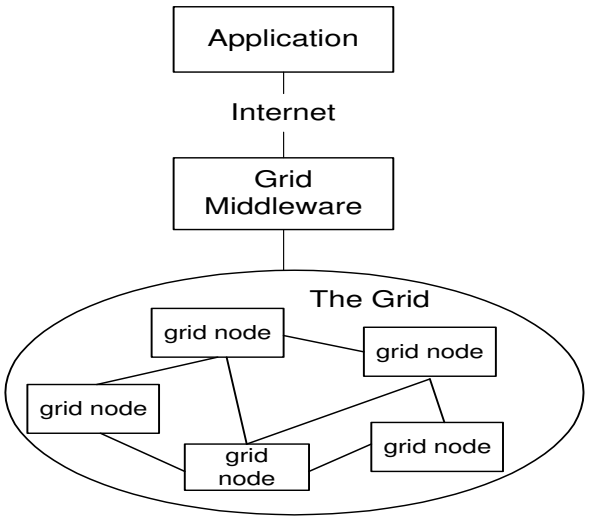


Figure 1: Component sketch: The grid

The grid has a hetrogeneous nature, spanning multiple administrative domains. This means that it is not an entirely stable environment, at least not as stable as a single computer of which the user has total control. There are as yet no guarantees that your job will not be canceled or receive very low priority on behalf of local users or even other grid users. The applications using the Grid should be aware of this so that the appropriate action, for example resubmission or compensation can be taken.

The Application

The parameter studies performed is as stated before a good example of a task which is easily parallelizable on the grid since the different simulations can be made independent of each other. Parameters sweeps can also fairly easily be extended using optimizer routines so that a models parameters can be optimized for a given criterion, increasing the usefulness of parameter studies.
As for the application itself, the user can specify model parameters, the range and the step size use. The job will then be submitted to the grid for execution and the results will be returned to the user at his request when the jobs have finished. The resulting plot can be viewed with the plotting utility of the users choice. Job monitoring is supported automatically through the Nordugrid middleware, both in detailed format about every job and as an overview through the grid monitor depicted in figure 3.

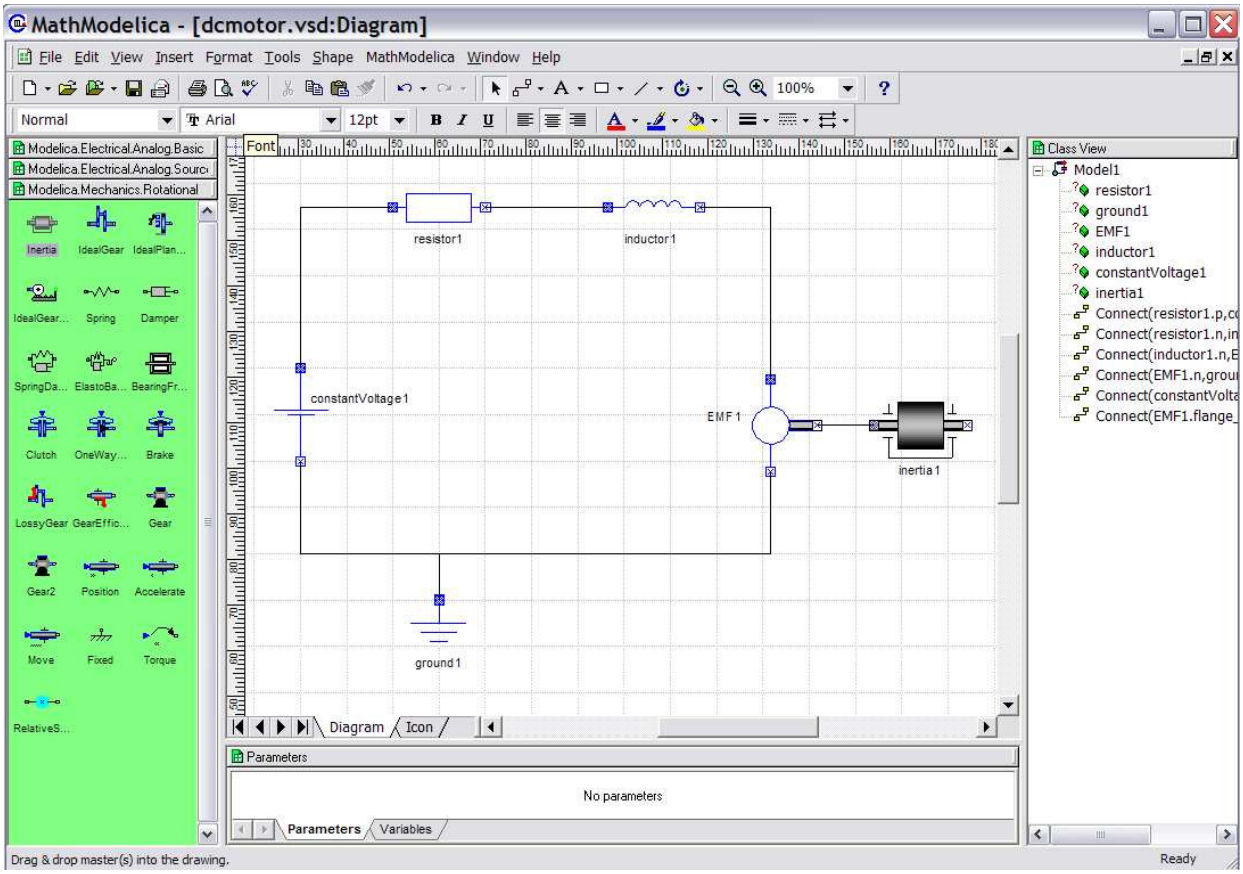


Figure 2: The MathModelica graphical model editor from MathCore Engineering [9]

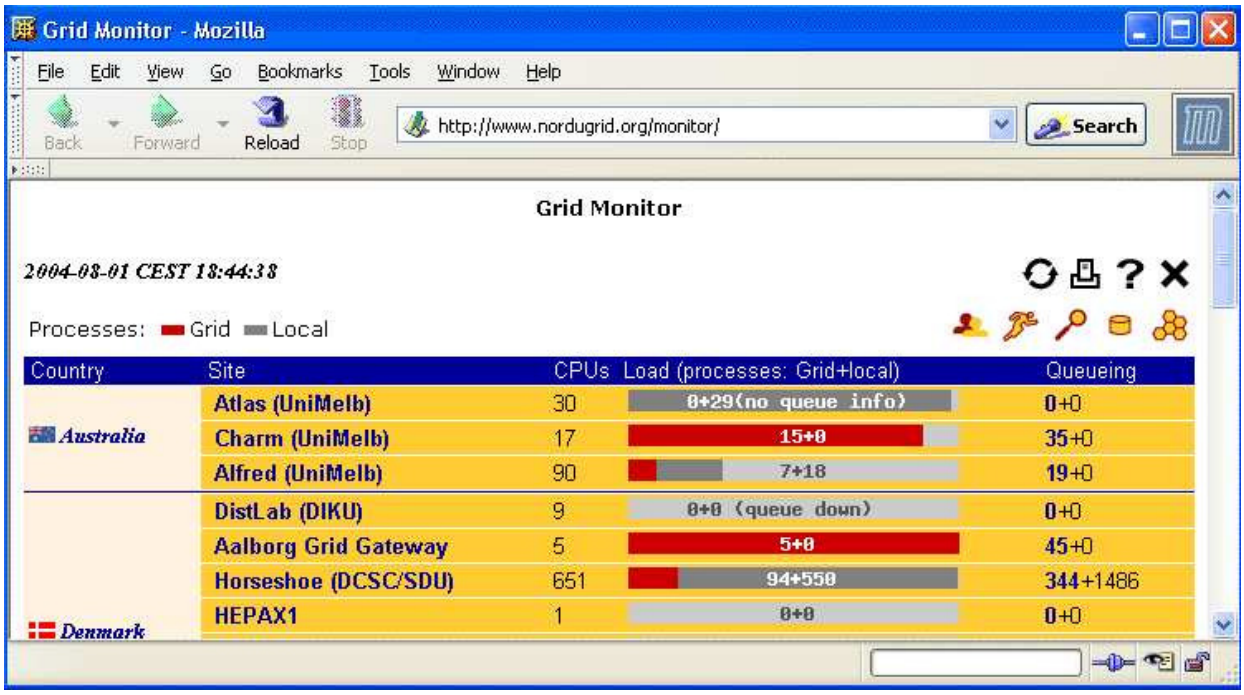


Figure 3: The Nordugrid Grid Monitor

Example	Timesteps	N.of parameter values	Exec. time, single	Exec. time, Nordugrid.
DCMotor, 40 eqns	10000	10	0.5m	10m
DCMotor, 40 eqns	10000	100	4m	175m
DCMotor, 40 eqns	1000000	10	7m	29m
DCMotor, 40 eqns	1000000	100	68m	200m
Twoloops, 1016 eqns	1000	10	35m	24m
Twoloops, 1016 eqns	1000	100	330m	75m
Twoloops, 1016 eqns	10000	10	423m	59m
Twoloops, 1016 eqns	10000	100	4229m	560m

Table 1: Time to execute different parameter studies on the Nordugrid compared to a single machine.

What is done in the application is in short this:

1. Initialization of the environment, including setting up of the Nordugrid proxy and credentials for accessing the grid.

2. Generation of submission and result retrieving scripts.

3. Generation of Extendend Resource Job Specification, XRSL-file. The XRSL-file specifies the nature of the job, such as files which should be included, command line arguments and the estimated resources required.

4. Compilation of the Modelica model

5. Merging of results into comprehensible format.

What the user essentially has to do is first to write the model in question, using a text editor or a visual tool such as MathModelica depicted in Figure 2. The user also has to specify the parameters, the range to investigate and the step size in a simple and straightforward text file.
Some additional software is required for usage:

1. A modelica compiler, such as the OpenModelica compiler [10, 11] or Dymola [12].

2. A C-compiler, for example gcc.

3. The Nordugrid toolkit [4]

4. Certificates authorizing usage of the appropriate grid resources.

GridParamSweep can be downloaded from the GridModelica Project homepage [7].

Results and Conclusions

Not surprisingly the grid proves very well suited for this type of applications, especially if the job itself is fairly complex. As we are dealing with independent tasks, the speedup compared to doing the same job on one workstation is almost proportional to the number of steps in the parameter sweep. We have successfully tested different models generating systems with up to a thousand equations. The number of processors used are roughly equivalent to the number of steps in the parameter sweep. Exact results are found in table 1. The first example is the circuit depicted in figure 2 and the second example is from the Modelica Multibody library [8].
Standalone simulations in table 1 were conducted on a Pentium III 1.8 Ghz with 512 Mb of RAM. Grid simulations in table 1 are averages from five simulation runs per test case, conducted on the Nordugrid in five consecutive days, 2004-07-14 2004-07-19 between 10.00 and 14.00. It is important to note that execution time may vary greatly with the current load of the grid. Some jobs may be placed in queue, some may receive low priority and some may be scheduled for execution on a comparatively slow cluster node. Even so, results show that all but the smallest parameter studies of simulations benefits from grid deployment. As the overhead for job submission is fairly large the greatest speedup compared to single machine execution are found when the job is fairly complex. The DCmotor model proves too small for efficient grid execution.
It should be noted that the Twoloops model with its 1016 equations is by no means a large model. Tests of the GridParamSweep application have been carried out on models consisting of way beyond 20000 equations, producing over 3GB of data with no scal-

ing problem whatsoever.

Discussion

The most serious problem encountered in this study was the large overhead for submitting jobs to the grid, limiting its usefulness to fairly complex simulations. This is not really that surprising since most grid jobs today are extremely computationally expensive physics simulations and data analysis jobs. Their normal execution time may very well exceed 12 hours and if that is the case, an overhead of half a minute per job submission is really not a problem.

The average overhead for each job submission varies with the grid load and the connection speed of the submitting machine but never exceeded 60 seconds in our measurements. This overhead is however not quite cumulative with the number of jobs since the major part of this delay normally is due to the grid manager program and on the assigned grid node rather than on the submitting workstation, leaving the user free to submit the next job rather than wait for the previous one to start it's execution. Recent grid middleware development has reduced this overhead and further improvement can also be expected as the number of smaller grid jobs increases. It is also possible to reduce the overhead significantly by manual tuning of the grid submission procedure, for example avoid checking some grids that are known to be slow or not operational.

This overhead still means though that grid execution is not so useful for small simulations spanning few parameter steps. It is difficult to give estimations on when a job should be submitted to the grid and not since so many factors are involved which affect the execution time and the overhead. What can be clearly stated is that simulation of complex models no longer requires a top-of-the-line workstation closely at hand. Instead, the huge amount of spare cycles the grid has to offer can be used.

Another problem which is far more difficult to solve is the problem of moving data. The Twoloops example with 10000 time steps produces approximately 257 MB of data. When executing on a single machine, moving the data is of course not an issue since the data is produced directly on the workstation in question. When executing the job on a grid node however, the data is produced somewhere else. A high speed network is thus quite useful for retrieving the data efficiently. In our test runs, we used a mere 5 Mbit non dedicated ethernet connection to the internet. This means that a significant decrease in data retrieval time should be obtained with a better internet connection.

The presentation of the raw results presents a bit of a problem. In general, the user is interested in how the behavior of a variable x varies with time depending on how a parameter p is chosen. In the dcmotor in Figure 2, one would perhaps be interested in the rotation angle of the inertia in relation to what value is chosen for R on the resistor. This means that ordinary 2d plots are not always sufficient to present the whole result, at least not if the number of parameter values studied is large. The plot will have to be extended to three dimensions using some simple Matlab commands.

In conclusion, grid computing indeed has a lot to offer the modeling community in conjunction with high level modeling languages such as Modelica. Many of the typical jobs submitted to the grid today are different types of simulations but they are to a great extent specially written in C or Fortran and adapted for the grid by hand which is a difficult and time consuming task. High level modeling in GridModelica should considerably facilitate the process of cheaply simulating computationally expensive tasks in the future.

Future work in GridModelica

We would like to emphasize that this is by no means a complete grid simulation environment yet but merely an example of how the grid can transparently be used by modelers even though the grid is still comparatively young.

There are quite a number of other improvements to the GridParamSweep application that could be conceived. A few of them we have considered are:

- Integration with the OpenModelica compiler in order to make the application completely open source.
- Multiple submissions of the same job in order to reduce chances of a total job failure. At least one instance of the same job should always execute successfully.
- Automatic adaptive job submission, which could greatly reduce overhead.

- Clustering of smaller jobs into one job, thus increasing the applications usefulness for smaller jobs by reducing the number of submissions that has to be done.

- Better checking of job termination and acting accordingly. If a job seems to have failed, an immediate resubmission should be done.

Some of these may eventually make their way into the GridModelica framework but it is actually not likely that they will be implemented in the GridParamSweep application.
Future work on the GridModelica framework will further facilitate simulation deployment on the grid and will also include features such as language extensions for high level parallelization, internal automatic parallelization and job scheduling for the grid.

Related Work

There are a number of other tools for conducting parameter studies, some even in parallel though none has to our knowledge yet been adapted for computational grids. Two applications that treat Modelica models are the MOPS framework [13] and the Distributed Parameter Study application [14]. They have in common that they enable distributed parameter studies but they also require direct access to the computational nodes, for example via rsh, which the grid for security reasons does not allow. They also require considerably more setup on the remote computing nodes. This makes them perhaps more reliable to use but also less powerful and more expensive due to the necessary investment in hardware compared to parameter sweeps performed on the grid.
Automatic parallelization of Modelica models can also be performed on a per-model basis on a much more fine grained level as done by Peter Aronsson [15]. This approach will in the future be merged into GridModelica, adding low level parallelization to the framework.

References

[1] Fritzson P. *Principles of Object-Oriented Modeling and Simulation with Modelica 2.1*. Wiley-IEEE Press, 2004.

[2] Elmqvist H, Mattson S-E, Otter M. *Modelica-A Language for Physical System Modeling, Visualization and Interaction*. In Proceedings of the IEEE Symposium on Computer Aided Cibtrol System Design, Hawaii, USA, August 22-27 1999.

[3] Foster I, Kesselman C, editors. *The Grid - Blueprint for a New Computing Infrastructure*. Morgan Kaufmann Publishers, 1998.

[4] *The Nordugrid Virtual Organization*. http://www.nordugrid.org.

[5] *The Globus Alliance*. http://www.globus.org.

[6] *The Swegrid Computational Resource*. http://www.swegrid.se.

[7] *The GridModelica homepage*. http://www.ida.liu.se/labs/pelab/modelica/GridModelica.html.

[8] Otter M, Elmqvist H, Mattson S-E. *The new Modelica Multibody Library* proceedings of the Modelica 2003 conference, November 3-4 2003, Linköping Sweden.

[9] *MathCore Engineering*. http://www.mathcore.com.

[10] *The OpenModelica project*. http://www.ida.liu.se/~pelab/modelica/OpenModelica.html.

[11] Fritzson P, Aronsson O, Bunus P, Engelson V, Johansson H, Karström A, Saldamli L. *The Open Source Modelica Project*. In Proceedings of the 2nd International Modelica Conference, Munich Germany, 18-19 2002. http://www.ida.liu.se/~pelab/modelica/OpenModelica.html.

[12] *Dynasim*. http://www.dynasim.com.

[13] Joos H-D, Looye G, Moormann D. *Design of Robust Inversion Control Laws using Multi-Objective Optimization* AIAA Guidance and Control Conference, Montreal, Canada, 2001.

[14] Engelson V, Fritzson P. *A Distributed Simulation Environment* Proceedings of SIMS 2002 conference.

[15] Aronsson P. *Automatic parallelization if Simulation Code from Equation Based Languages*. Licenciate thesis No. 933, Department of Computer and Information Science, Linköping University, Sweden

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

THE USE OF A ZERO DIMENSIONAL MODEL IN ORDER TO CHARACTERISE KNOCK IN SPARK IGNITION GAS ENGINES

Eric Ollivier, Jérôme Bellettre, Mohand Tazerout
Ecole Des Mines de Nantes, DSEE
4, rue Alfred Kastler, B.P. 20722
44307 Nantes Cedex 3 , France

Abstract

This paper focuses on the modelling of the exhaust gas temperature of a spark ignition engine in order to propose a new and non-intrusive method of knock detection. A zero-dimensional model is developed and accounts for the heat transfer amplification due to knock. The heat transfer coefficient is a function of the mass burnt rate because knock intensity is linked to the autoignited mass of fuel. The decrease of exhaust gas enthalpy du to knock is pointed out and analysed for a large tunings and fuel composition range. The numerical results show that a knock indicator based on the calculated maximal temperature and the measured average exhaust gas temperature as well as engine tunings can be developed.

Nomenclature

c_p Specific heat at constant pressure [J/kg.K]
DKI Dimensionless knock indicator [ ]
$\frac{dx_b}{d\theta}$ Mass burnt rate [CA^{-1}]
E Total energy [J]
EVC Exhaust valve closing [CA]
EVO Exhaust valve opening [CA]
H Heat transfer coefficient [W/m^2K]
IMEP Indicated mean effective pressure [Pa]
L Time duration of pressure oscillation recording [CA]
MN Methane number
$\dot{m}_a$ Inlet gas mass flow rate [kg/CA]
$\dot{m}_e$ Exhaust gas mass flow rate [kg/CA]
$\dot{m}$ Total mass flow rate [kg/CA]
N Engine speed [$rev.min^{-1}$]
p In-cylinder pressure [Pa]
p_0 In-cylinder pressure at the end of the compression stroke [Pa]
$\tilde{p}$ Filtered in cylinder pressure [Pa]
r Volumetric efficiency [ ]
$\hat{R}$ Universal constant of gas [J/kg.K]
T Gas temperature
TDC Top dead centre
T_{exh} Averaged exhaust gas temperature [K]
T_{max} Maximal gas temperature [K]
W Work [J]
x_b Burnt mass fraction [ ]

Greek Letters
α Spark timing [CA]
α_{cr} Critical spark timing [CA]
$\Delta\theta$ Combustion duration [CA]
ϕ Equivalence ratio [ ]
ρ Density [kg/m^3]
θ Crank Angle [CA]
θ_0 Beginning of the combustion [CA]
θ_i Time of pressure oscillation appearance [CA]

Subscripts
c_r critical
n normal
k knocking

1- Introduction

Knock is an abnormal combustion in the cylinder of SI engines. It is the result of the autoignition of one part the end-gas because of their sudden rise in temperature and pressure due to the propagation of the primary flame front. The detonation of this fraction of mixture leads to pressure waves characterized by their oscillatory frequency (generally comprised between 5 and 10 kHz). These waves reflect on the cylinder walls, deteriorate the thermal boundary layers and generate high heat flux [1-5]. This phenomenon is very undesirable because it leads to an increase in the wall heat losses to the detriment of the output of the engine. Moreover, the engine is submitted to high mechanical efforts and thermal loads which in short term can lead to its destruction.

Nowadays, as the compression rates of the engines tend to increase in order to reach economical and environmental concerns, the manufacturers are faced with this problem. It is known that with liquid fuels this phenomenon can be contained [6], but it is not always the case with gaseous fuels because their properties are highly variable. Some technologies exist to detect the phenomenon such measuring the cylinder vibrations by accelerometry or by the measurement of the in–cylinder pressure during the combustion. The latter is the more reliable but is intrusive and harmful for the engine. An alternative must be found to detect knock easily.

As already mentioned, the energy balance of the engine is modified by knock : the efficient work of the engine and the exhaust gas enthalpy decreases because of the amplification of the wall heat losses. Abu-Qudais [7] claimed that it is possible to define a knock indicator based on the measurement of the exhaust gas temperature by defining a threshold value under which knock must have appeared. This value is highly variable with the engine tunings or fuel quality so this process needs the construction of a performance map which would be available only for the considered engine. Thus, the generalization of such a method of knock detection requieres a new knock indicator. The present paper presents the development of this new indicator.

In this study, a zero dimensional model was developed in order to describe the thermodynamic cycle of a spark ignition engine running with knock.

Knocking combustion is described by a double Wiebe law since during knock two types of combustion coexist: a deflagration (normal combustion) and a detonation induced by the autoignition of one part of the end gas. The correlation of heat transfer used in this model is a function of the mass burnt rate [8]. The model is calibrated on the basis of experimental measurements of in-cylinder pressure during normal and knocking combustion. The exploitation of the model aims at linking knock intensity to engine tunings and exhaust gas temperature.

The paper is divided into 2 main parts : after explaining the modeling assumptions, the model exploitation is exposed and discussed.

2- Modeling Assumptions

A single zone, zero-dimensional model aims at the prediction of the cylinder pressure, the gas temperature, the mass flow rates, wall heat transfer during one engine cycle. The single zone model assumes that at any instant of the cycle, pressure and temperature are homogenous in the whole combustion chamber. No difference is made between the end gas and the burnt gas. Such model relies on the mass conservation principle (1), the first law of the thermodynamics (2), and the ideal gas law (3) :

$$\dot{m} = \sum_j \dot{m}_j \tag{1}$$

$$\dot{E} = \dot{Q} - \dot{W} + \sum_j \dot{m}_j h_j \tag{2}$$

$$p = \rho \hat{R} T \tag{3}$$

The inlet and exhaust mass flow rates are calculated assuming that the flow in the valves can be assimilated to an isentropic flow in a nozzle

The total heat losses during the cycle can be calculated from the area exposed to flux (which are a function of the piston position) and from a correlation for the heat tranfer coefficient (to be seen later).

Modeling of the knocking combustion

In a zero dimensional model, the combustion is only considered as an additional source of energy. As knock results in the coexistence between two modes of combustion (a deflagration and a detonation – the knock), a double Wiebe law (4) is used to model the combustion rate. The contribution of the two modes

of combustion is taken into account enabling to link the intensity of knock to the autoignited mass of end gas. In equation (4), *n* and *k* subscripts refer to the normal and knocking combustion respectively. The coefficients
K_n and K_k represent the percentage of mass burnt during each mode of combustion. For a total combustion $K_n + K_k = 1$ and $M_n = M_k = 6.908$. K_k increases with knock intensity and for a non knocking combustion K_k tends towards zero.

$$\frac{dx_b}{d\theta} = K_n A_n \frac{M_n+1}{\Delta\theta_n}\left(\frac{\theta-\theta_{0_n}}{\Delta\theta_n}\right)^{M_n} \exp\left[-A_n\left(\frac{\theta-\theta_{0_n}}{\Delta\theta_n}\right)^{M_n+1}\right]$$
$$+ K_k A_k \frac{M_k+1}{\Delta\theta_k}\left(\frac{\theta-\theta_{0_k}}{\Delta\theta_k}\right)^{M_k} \exp\left[-A_k\left(\frac{\theta-\theta_{0_k}}{\Delta\theta_k}\right)^{M_k+1}\right] \quad (4)$$

This method was applied by Brecq in 2002 [9] and enabled him to find good agreement between his model and experimental measurements regarding bunt mass fraction.

Heat transfer model

The correlation used for the instantaneous convective heat transfer coefficient calculation is given by equation (5) [8] :

$$H = 0.25N\left(\frac{p}{p_0}\right)^{0.35}\left(1+25\frac{dx_b}{d\theta}\right)^{2.5}\left(1+5.10^5\,\dot{m}_a\right) \quad (5)$$

Knock intensity and the resulting heat transfer amplification are linked to the portion of mass which autoignites. As this correlation is a function of the mass burnt rate, the effect of knock on heat losses during one cycle is taken into account.

Example of knock modeling

Experimental measurements of the in-cylinder pressure during both knocking and non knocking combustion [9,10] are used for the calibration of the model. The model is run for many engine tunings and natural gas quality. And the averaged error on the *IMEP* (the work per cycle over the compression and expansion strokes divided by the displaced volume) for all studied cases is comprised between 2 and 3%.
An example of simulation is shown in figures 1-4 for a natural gas supplied engine set at N = 1500 rpm, $\phi = 0.7$, $r = 0.82$. The volumetric composition of the natural gas in this case is given in Table 1. The corresponding methane number of this gas is 81.

CH_4	90.5
C_2H_6	7.1
C_3H_8	1.4
C_4H_{10}	0.5
C_5H_{12}	0.02
N_2	0.48

Table 1: Volumetric composition of the natural gas

In this case, a cycle is knocking if $\alpha > 28\,\text{CA}$ before TDC. This spark timing is called the critical spark timing and is noticed α_{cr}. Increasing this parameter enables knock occurrence.

Figures 1 (a, b) show that good agreement is found between experimental and simulated pressure by the conjugate use of equations (4) and (5). The modeled mass burnt rate relevant to figures 1 is represented in figure 2. The ignition delay which corresponds to the time development of the flame can vary from one engine tuning to another (7 CA if the spark timing is fixed at 20 CA BTDC, 13 if spark timing is 32 CA BTDC). In the particular case of $\alpha = 32$ CA, knock happens 11 CA after the beginning of the normal combustion and 43% of the introduced mass autoignites, which considerably reduces the time duration of the total combustion. Figure 3 represents the resulting instantaneous heat transfer coefficient from the burnt gas to the wall in the combustion chamber. As a result of the autoignition, two peaks can be observed which qualitatively is in good agreement with the measurements of Syrimis [4] and Enomoto et al. [2].

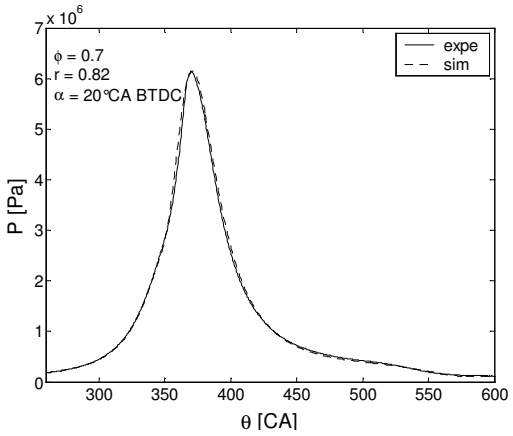


Fig. 1 (a): Comparison between measured and simulated pressure of a knock free cycle

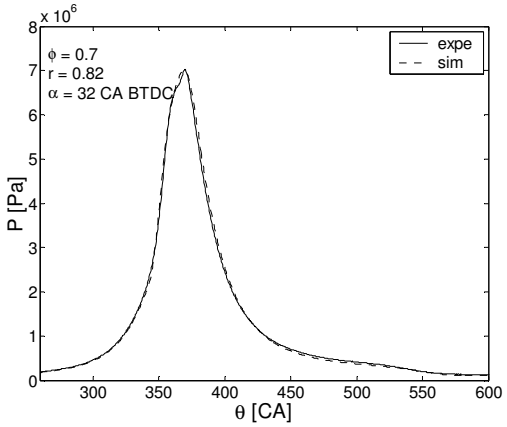


Fig. 1 (b): Comparison between measured and simulated pressure of a knocking cycle

It can be observed in figure 4 that the exhaust gas temperature is lower in the case of a knocking combustion du to the amplification of the wall heat transfer. On other hand, the maximal temperature of the gas during the cycle, T_{max}, is higher as the combustion is shortened. However, spark timing has an effect on T_{exh} and wall heat transfer similar to knock. Hence both effects are difficult to distinguish.

3-Exploitation

Knock indicator reference

For the following of this study, a reference knock indicator must be used to link the heat transfer amplification with the knock intensity. The most widely used indicators rely on the in-cylinder filtered pressure. They are the maximum amplitude of pressure oscillations (*MAPO*, eq. 6-a) [2, 4, 5] and the integral of modulus of pressure oscillations (*IMPO*, eq.6-b) [9]. The latter is a way to represent the energy contained in the high frequency oscillations of the cylinder pressure signal, which occurs due to knock. But both depends on the experimental setup (engine geometry and on the engine tunings) which make them not really universal.

Brecq et al. [10] defined a dimensionless knock indicator (*DKI*) based on the measurement and filtering of the in-cylinder pressure and which is a funtion of the *MAPO* and *IMPO*

.

$$MAPO = \frac{1}{N}\sum_{1}^{N} \max_{ST,ST+W} \left|\tilde{p}\right| \qquad \text{(6-a)}$$

$$IMPO = \frac{1}{N}\sum_{1}^{N} \int_{ST}^{ST+W} \left|\tilde{p}\right| d\theta \qquad \text{(6-b)}$$

$$DKI = \frac{IMPO}{MAPO \times W} \qquad \text{(6-c)}$$

N is the number of measured pressure cycles and W is the width of the computational window (here 60 CA).

A combustion can be considered as knocking if $DKI < 0.19$, whatever the engine tunings are. Figure 5 shows a graphical representation of the *DKI* in the case of a knocking cycle. The indicator will be used in following of the study as reference.

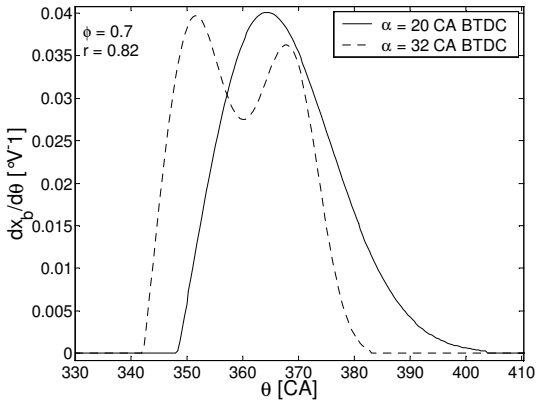


Fig 2: Calculated mass burnt rates for a knocking and a non knocking cycle

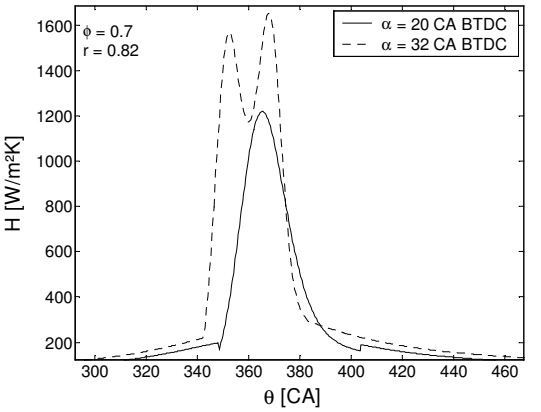


Fig. 3: Simulated instantaneous heat transfer coefficient during a non knocking cycle and a knocking cycle.

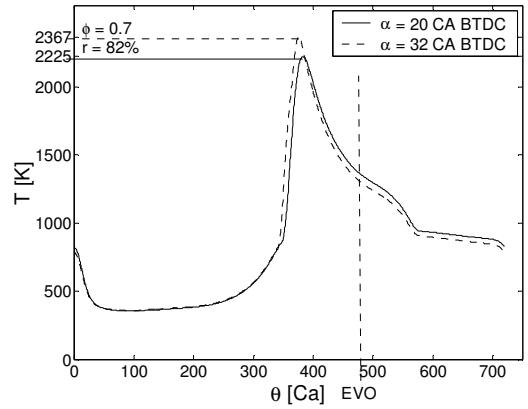


Fig. 4: Simulated gas temperature during a knocking and a non knocking cycle

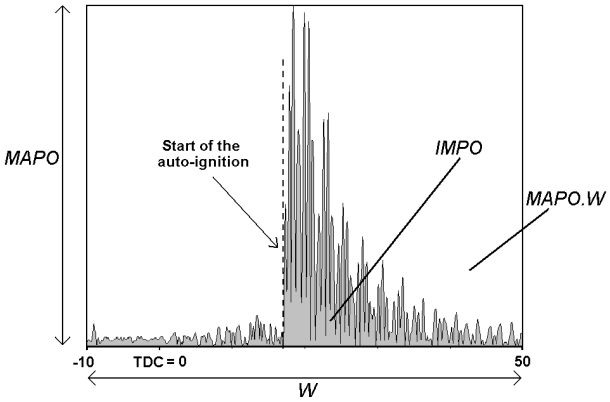


Fig. 5: *DKI* representation for a knocking cycle

Averaged exhaust gas temperature and knock indicator

On the basis of the simulated temperature of the gas during one cycle, the averaged exhaust gas temperature is calculated by equation (7):

$$T_{exh} = \frac{\int_{EVO}^{EVC} \dot{m}_e c_p T d\theta}{\int_{EVO}^{EVC} \dot{m}_e c_p d\theta} \qquad (7)$$

As already mentioned, the exhaust mass flow rate is calculated assuming that the flow in the valves can be assimilated to an isentropic flow in a nozzle. Details for the calculation can be found in [9].

Figure 6 represents the simulated averaged exhaust gas temperatures as a function of *DKI* for several engine tunings. Knock is obtained by advancing spark timing for a given equivalence ratio and volumetric efficiency. Knock is also generated by varying the rate of C_3H_8 from 1.4% to 4%, keeping the equivalence ratio constant (ϕ = 0.7). The methane number in this case becomes: *MN* = 76 against 81. Figure 6 shows that no threshold value available for every engine tunings can be defined to detect knock nor to give the intensity of the phenomenon (pts 1, 2, 3, 4). This is du to the fact that the absolute level of the temperature depends on the tunings and fuel quality.

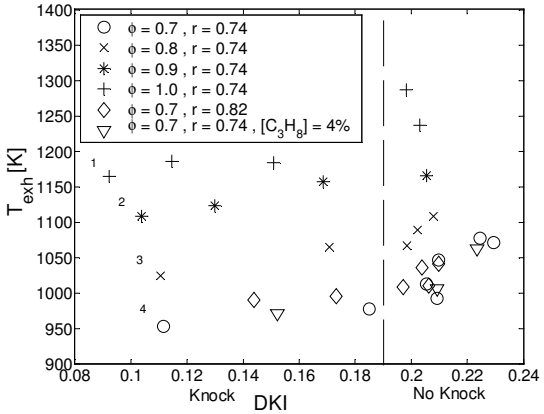


Fig. 6: Simulated averaged exhaust gas temperature as a function of knock intensity.

Definition of a new knock indicator

T_{max}-T_{exh} is plotted in figure 7. It can be noticed that a threshold value of this difference independent of the engine tunings exists (1280 K) under which the system can be considered as knock free can be defined.

The difference between $T_{max.}$ and T_{exh} is a good representation of the knock intensity and then can be considered as a new knock indicator.

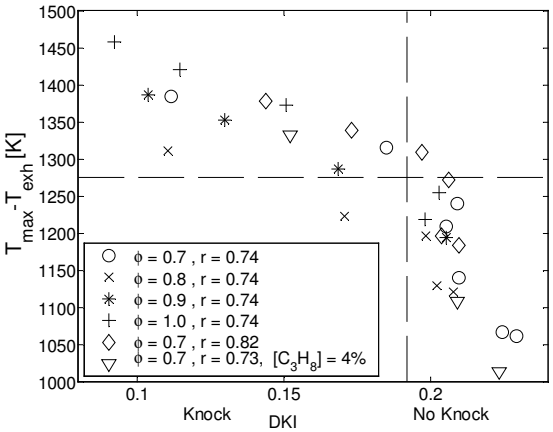


Fig. 7: Difference between T_{max} and T_{ech} as a function of knock intensity.

Modelling of the maximal temperature of the gas during the cycle

In practice T_{max} can't be measured easily. So it is of first importance to link T_{max} with T_{exh} because it is the only temperature which should be measured in order to make this indicator non intrusive in a future application.

It is known that the earlier the combustion starts in the cycle (because of the advancement of spark timing), the larger the work transfer from piston to the gas in the cylinder at the end of the compression stroke is. As a result, the instantaneous maximal temperature of the gas $T_{max.}$ grows with spark timing [11]. A reduction of the duration of the combustion induced by knock will have the same effect on $T_{max.}$.

As T_{max} grows the heat transfer from the gas to the wall are increased which tends to decrease T_{exh}.

The difference between those two temperatures should grow with spark timing mostly after knock is reached but the conjugate effects of spark timing and knock intensity on T_{max}-T_{exh} can't be easily discernable.

Figure 8-a shows the evolution of the ratio T_{max}/T_{exh} with spark timing, α. In the same way as T_{max}-T_{exh}, T_{max}/T_{exh} grows with α. Moreover, if the mixture equivalence ratio is enlarged, the duration of the combustion is shortened and the temperature of the gas is enlarged. This makes the critical spark timing α_{cr} lower and the rate of growth of T_{max}/T_{exh} higher. According to the results plotted in figure 8-a, T_{max}/T_{exh} can take the following form (8) :

$$\frac{T_{\max}}{T_{exh}} = a\alpha + b \qquad (8)$$

where a depends on the product $r \times \phi$ (Fig. 8-b) and b is fixed at a constant value 1.7. The product $r \times \phi$ can be directly related to engine consumption.

Equation (8) leads to a simple expression (9) of T_{max}-T_{exh} independent of the composition of the natural gas (in the tested range where $[C_3H_8] \leq 4\%$):

$$T_{\max} - T_{exh} = \left[K_1 r\phi\alpha + K_2\alpha + K_3\right]T_{exh} \qquad (9)$$

The values of K_1, K_2, K_3, are in this particular case 0.078 [CA^{-1}], -0.025, 0.7 [CA^{-1}] respectively.

Figure 9 shows that, except for three points, T_{max} calculated by (6) is over or under estimated by a maximum of 100 K compared with the actual value. This is quite satisfactory because T_{max} is comprised between 2100 K and 2600 K if the equivalence ratio of the mixture vary from 0.7 to 1.0.

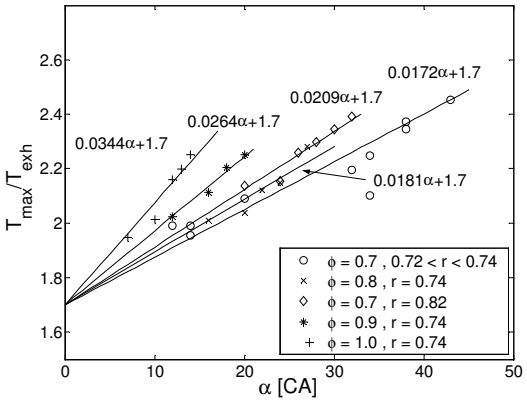


Fig. 8-a: T_{max}/T_{exh} ratio as a function of spark timing.

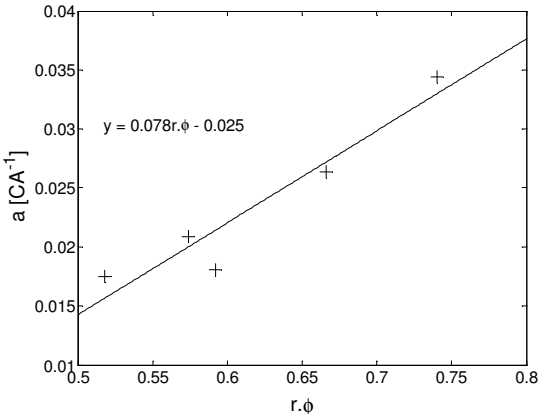


Fig. 8-b: Slope of the curves $T_{max}/T_{exh} = f(\alpha)$ as a function of $r \times \phi$

Figure 10 shows that the value of new knock indicator, defined by equation (9), under which the system would be knock free is 1265 K. This value is relatively close to the value precedently given. Moreover, because of the error introduced by equation (8) and in, a future application, by the exhaust gas temperature measurement, a range of error of at least ± 15 K should be set. Thus, the knock threshold value for the non intrusive knock indicator should be imposed smaller (1250 K).

4-Conclusion

A zero dimensional model was developed for the modeling of the heat transfer from the burnt gas to the wall during knock in spark ignition engine. The effect of the increase of the heat losses on the averaged exhaust gas temperature is pointed out and it was shown that the measurement of the latter is not sufficient to detect knock whatever the engine settings are. A new knock indicator was defined. For a given mixture equivalence ratio, volumetric efficiency and spark timing, the engine will be knock free if the averaged exhaust gas temperature

follows the rule (10) :

$$\left[K_1 r\phi\alpha + K_2\alpha + K_3\right]T_{exh} < 1265K \quad (10)$$

This criteria should be extended to strongest variation of methane number of the fuel.

The extension of this indicator to another engine will require the measurement of the exhaust gas temperature and of the cylinder pressure. This is necessary to determine the values of the K_1, K_2, K_3 coefficients and the value of threshold value before being applied in an industrial way.

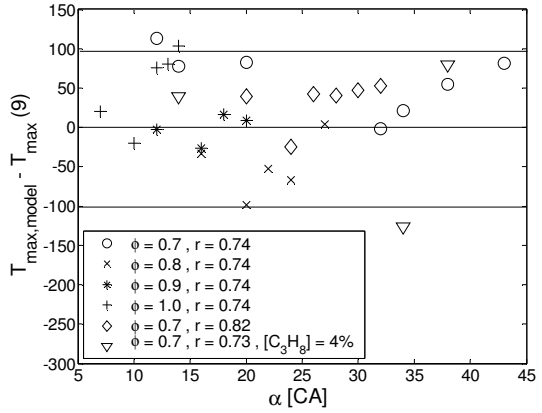


Fig. 9: Difference between given by the single zone model $T_{max,model}$ and $T_{max,(9)}$ calculated by (9).

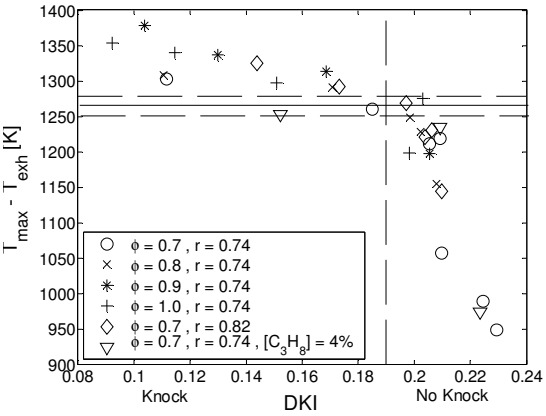


Fig. 10: Difference between T_{max} and T_{exh} given by equation (9).

Bibliography

1. Douaud, A., *Eléments d'analyse du cliquetis et de ses effets.* Revue de l'Institut Français du Pétrole, Vol 38, pp 665-674, 1983.

2. Enomoto,Y., Kitahara, N. and Takai, M., *Heat losses during knocking in a four-stroke gasoline engine*. JSME International Journal , Series B, 37, 1994

3. Lu, J. H., Ezekoye, D.,. Iiyama, A., Greif, R. and Sawyer R. F., *Effect of knock on time resolved engine heat transfer. SAE paper 890158*

4. Syrimis, M., *Characterization of knocking combustion and heat transfer in a spark ignition engine.* Ph. D. Thesis, University of Illinois, 1996.

5. Chun K. M, Heywood J.B, *Characterization of knock in a spark ignition engine*. SAE Technical Paper, (890156), 1989.

6. Moses, E., Yarin, A. L. and Bar-Yoseph, P., *On knocking prediction in spark ignition engines*. Combustion and Flame, Vol. 101, pp 239-261, 1995.

7. Abu-Qudais, M , *Exhaust gas temperature for knock detection and control in spark ignition engine.* Energy Conversion Management, Vol. 37, No 9, pp 1383-1392, 1996.

8. Trapy, J , *Modeling a heat-exchange coefficient specific to spark-ignition engines and taking running with knock into consideration.* Moteur d'automobile à allumage commandé dans le nouveau contexte européen. Journées internationales, Aix-La-Chapelle, 23-25 oct. 1985, Comptes rendus, p. 9.1-9.29.
IFP

9. Brecq, G, *Contribution à la caractérisation thermodynamique du cliquetis dans les moteurs à gaz : apllication à de nouvelles méthodes de détection*, PhD thesis , Ecole des Mines de Nantes, 2002.

10. Brecq, G, Bellettre, J, Tazerout, M, *A new indicator for knock detection in combined heat and power engines,* International Journal of Thermal Sciences, Vol. 42(5), 2002.

11. Heywood, J.B. *Internal combustion engine Fundamentals*. McGraw-Hill Book Company, 1988.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

AN INTEGRATED FRAMEWORK FOR MODEL-DRIVEN PRODUCT DESIGN AND DEVELOPMENT USING MODELICA

Adrian Pop[1] Olof Johansson[2] Peter Fritzson[3]
Programming Environments Laboratory
Department of Computer and Information Science
Linköping University
58131 Linköping
Sweden

Abstract

This paper presents recent work in the area of model-driven product development processes. The focus is on the integration of product design tools with modeling and simulation tools. The goal is to provide automatic generation of models from product specifications using a highly integrated set of tools. Also, we provide the designer with the possibility of selecting the best design choice, verified through (automatic) simulation of different implementation alternatives of the same product model. To have a flexible interaction among various tools of the framework an XML representation of the Modelica modeling language called ModelicaXML is used. For efficient search in a large base of simulation models the Modelica Database was designed.

1 Introduction and Related Work

Designing products is a complex process. Highly integrated tools are essential to help a designer to work efficiently. Designing a product includes early design phase product concept modeling and evaluation, physical modeling and simulation and finally the physical product realization. For conceptual modeling and physical modeling and simulation available tools provide advanced functionality. However, the integration of such tools is a resource consuming process that today requires large amounts of manual, and error prone work. Also, the number of physical models available to the designer in the product concept design phase is typically quite large. This has an impact on the selection of the best set of component choices for detailed product concept simulation.

To address these issues we have integrated new product concept design tools with physical modeling and simulation tools in a framework for product design. In our proposed framework, the product concept design phase of the product development process is based on Function-Means tree decomposition [7, 13]. This phase is implemented in a first version of a prototype tool called FMDesign, developed in cooperation with the Machine Design Group led by Petter Krus, IKP, Linköping University.

As an example of Function-Means tree decomposition we give a landing function in an airplane. This function can be represented by two different means: hydraulic landing gear or electric landing gear. Each of the two alternatives can be selected and configured to simulate its properties.

Starting from FMDesign tool, our integration work extends the framework in two ways:

- Providing a S*election and Configuration Tool* that helps the designer to choose a specific implementation for the means in the function-means tree from a Modelica model/

[1] Phone: +46 13 285781, Fax: +46 13 284499, Email: adrpo@ida.liu.se
[2] Fax: +46 13 284499, Email: olojo@ida.liu.se
[3] Phone: +46 13 281484, Fax: +46 13 284499, Email: petfr@ida.liu.se

component database. This tool also provides component configuration and has links to a Modelica standard based simulation environment for component editing.

- Providing an A*utomatic Model Generation Tool* that helps the designer to choose the best implementation from different design choices by evaluation through simulation of automatically generated models of candidate product concepts. If the designer is not pleased with the results, he/she can either implement new models for the components that did not perform in the desired way or reiterate in the design process and choose other alternatives for implementing different functions in the product, or change the configuration parameters for models at deeper levels of detail.

The paper is structured as follows: The next section presents an overview of our proposed framework. Section 3 enters in the details of the framework components and their interaction. Section 4 presents our conclusion and future work.

The presented system has similarities with the Schemebuilder tool [8]. However our work is more oriented towards the design of advanced complex products that require systems engineering, and targeted to the simulation modeling language Modelica, which to our knowledge has more expressive power in the areas of our research, than many tools for systems engineering that are currently widely used. For details on Systems Engineering, see [2].

2 Architecture overview

The architecture of our extended framework is presented in Figure 1. The entire product concept design process is iterative.

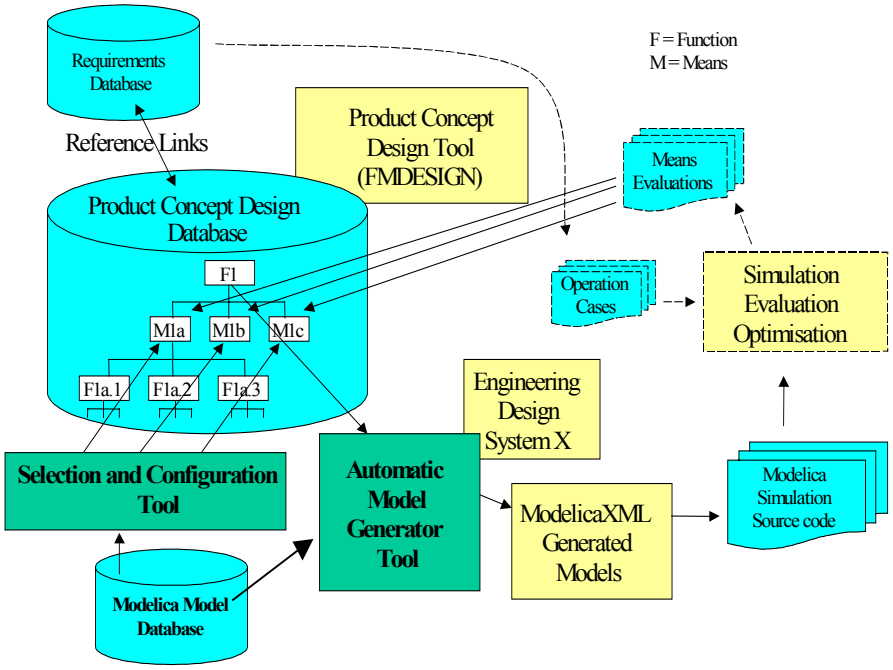


Figure 1: Design framework for product development

Starting from requirements for a product the designer will use the FMDesign prototype for modeling alternative product concepts. The knowledge base for designing a product is organized into function-means trees. A function in the product can be realized by alternative means. A product concept is a set of means that document selected solution alternatives for implementing the functions in a product concept. Example of a function is `"Actuator Power Supply"`, with means `"Hydraulic Power Supply"` or `"Electrical Power Supply"`. Means must be implemented by (physical) components arranged in a bill-of-material like tree of implementation objects.

One can roughly say that a means and its implementation are the same, but at different levels of detail. Implementation objects (not

shown in the figure) may represent existing component products on the market or manufactured components. Implementation objects carry data that is important for the product concept design, and references to more detailed design information like CAD-drawings, simulation models etc. Some (physical components) may implement several means, like an aircraft wing that creates lift and stores fuel.

To map suitable simulation model implementations to a means, the designer would use the Modelica Database query facility provided by the *Selection and Configuration Tool*. This tool also provides configuration of the simulation components and uses the desired Modelica environment for component editing.

When the product concept design phase of the product is sufficiently complete, the designer can generate code for simulation from the implementation tree using the *Automatic Model Generator Tool*. The generator will output models (different versions for different product concepts) in ModelicaXML. From Modelica-XML the models are translated to Modelica to be simulated. The designer can review the simulation results in tools like MathModelica [3], Dymola [1] or OpenModelica [10] and then selects (in FMDesign) the desired model alternative for the implementation. If the designer sees that some means do not perform in the desired way, a customized simulation model can be built, or a search conducted for more alternatives for that specific means.

3 Detailed framework description

In this section we present the tools from our proposed framework. Also, we briefly explain in each section how they interact.

3.1 ModelicaXML

Modelica [4, 9] is an object-oriented language used for modeling of large and heterogeneous physical systems. For modeling with Modelica, commercial software products such as MathModelica [3] or Dymola [1] have been developed. However, there are also open-source projects like the OpenModelica Project [10].

Modelica is translated to ModelicaXML using a Modelica parser (Figure 2).

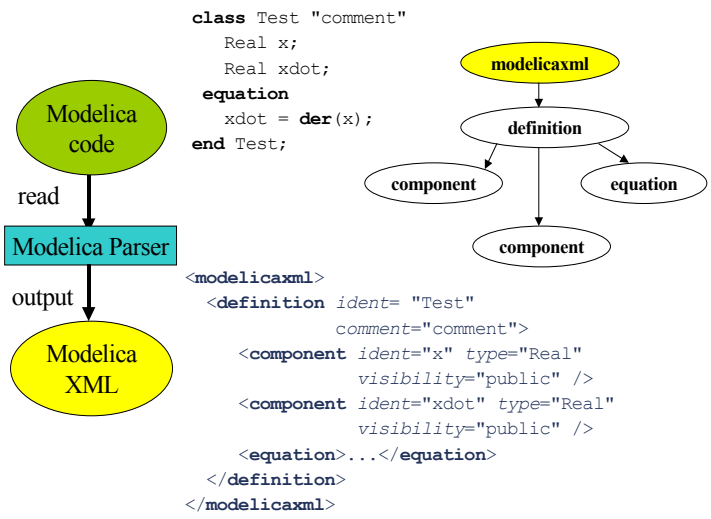


Figure 2: Modelica and the corresponding ModelicaXML representation

ModelicaXML represents an XML serialization of the Abstract Syntax Tree of the Modelica language obtained after the parsing. In our framework, ModelicaXML is used as an interchange format between the different design tools.

The advantages of having an alternative representation for Modelica in XML are:

- Flexible interaction and translation between different types of physical modeling languages and modeling tools. Also, easy generation of model documentation.
- Basic search and query functionalities over models.
- Easy transformation and composition of models [12].

For more information on ModelicaXML the reader is referred to [11] and [9].

3.2 Modelica Database (ModelicaDB)

The features of the Modelica language and Modelica tools has made easy for designers to create models. Also, the Modelica community has a growing code-base. In order to cope with interoperability between Modelica and other modeling languages we first developed ModelicaXML. However, scalability and efficient search features for XML require extensive skills in vendor specific products. To quickly get such features without taking on that huge learning

effort, we have designed the Modelica Database (ModelicaDB).

The Modelica Database is populated with Modelica models and libraries by importing their ModelicaXML representation. The UML model of this database is presented in the Appendix. For paper space reasons we use a somewhat customized compressed graphical representation of UML class diagrams, where inheritance is represented with a box between the class name and attributes box, where inherited super classes are preceded with a "->". For details on UML see [6].

Here we briefly explain the most important structures. They are tightly coupled with the Modelica structure [9, 11]:

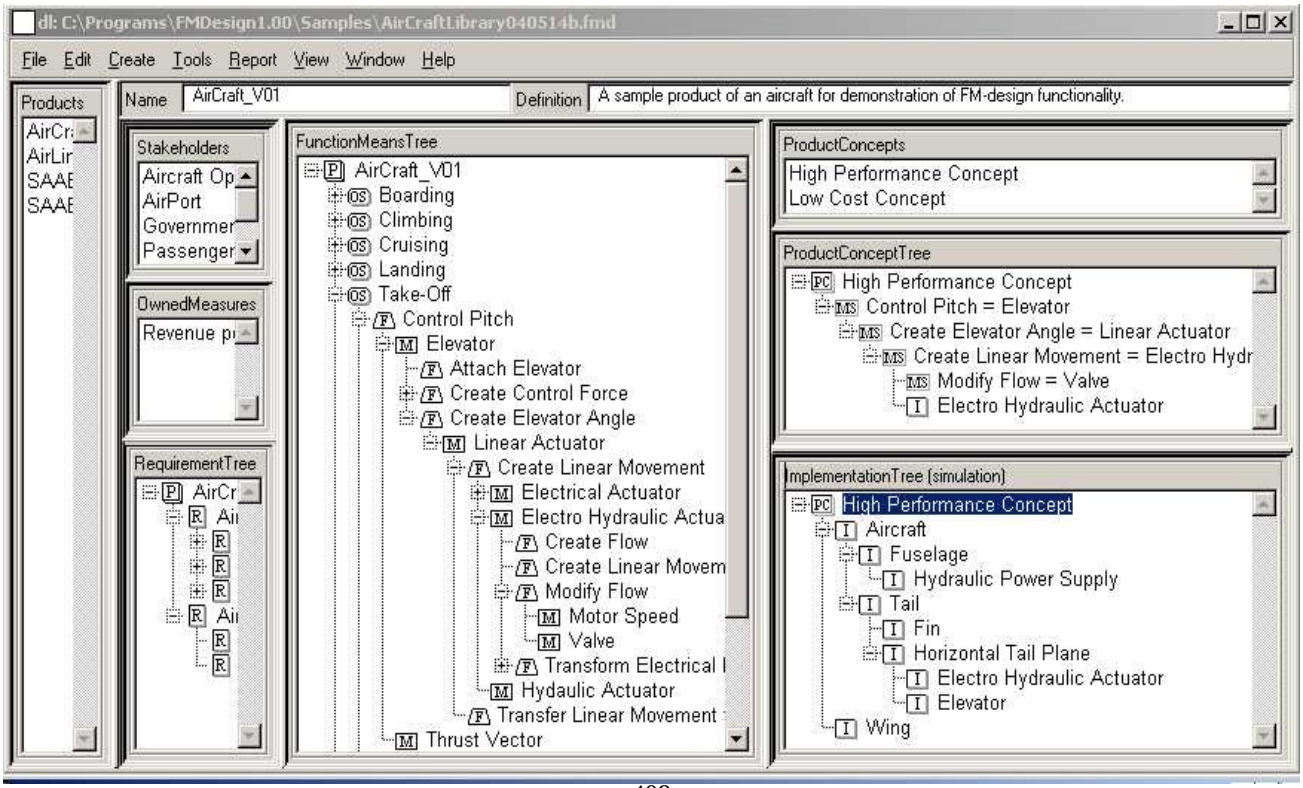
- *Modelica Repository*: contains several Modelica Models.
- *Class*: A class represents the fundamental model element from the Modelica language. It can include several *Component* clauses, *Equation* and *Algorithm* statements. The component sections can be declared as public or private in order to provide only the desired interface to the outer world. Specifying that the equation or algorithm sections are only active at the initialization phase they can be declared as initial.
- *Component*: used to define parameters, variables, constants, etc to be used inside a class.
- *Equations and Algorithms* are used to specify the desired behavior for a class.

In the product design framework the role of ModelicaDB is to provide searching and organization features of a large base of simulation models. This base grows with every product model developed or with the import of additional simulation models from other sources (i.e. the Modelica community). For example, if we want to obtain all the models that have certain parameter names we have to search in the database for all classes that have a component with the attribute `variabilityPrefix` set to `"parameter"` and has the specified name. These searches will be integrated in FMDesign using dialogs and completely transparent for the user.

3.3 FMDesign

The FMDesign (Figure 3) prototype tool helps the designer in creating product specifications using function-means trees.

Figure 3: FMDesign



The created product model is stored in a product design library for later reuse. Throughout the product concept design process the designer can use the existing concepts stored in the product design library in order to model the desired product. A somewhat simplified meta-model of the information structure edited in FMDesign is presented as an UML class diagram in the appendix section.

In the framework, FMDesign is the central front-end to specific components. FMDesign delegates searches in the ModelicaDB using the *Selection and Configuration Tool* and it uses the *Automatic Model Generation Tool* to generate the models for simulation.

As we can see in Figure 3, the work area is divided into several parts:

- *Products*: Here products are created, deleted and selected. When a product is selected, the trees owned by it and described below, are displayed.
- *Requirements Tree*: in this view the requirements for a product can be specified.
- *Function-Means Tree*: in this view the designer can define the operation states, functions, their alternative means etc, of the selected product.
- *Product Concepts*: Allows creating, deleting and selecting product concepts.
- *Product Concept Tree*: displays the currently selected Product Concept Tree, and allows the user to select which means that will implement different functions in the product, using drag-drop. Selected means can be customized for the current product concept by overriding the default values for its design variables owned by a selected means.
- *Implementation Tree*: displays and provides functionality for editing one of many configurable Implementation Trees for the currently selected product concept. These implementation trees organize the implementation objects that represent and refer to more detailed models of physical objects, functional models, simulation models, geometrical layout models etc, and organize them into trees that are useful for interfacing with tools later in the product development process.

We only use the Implementation Tree of type simulation to generate the Modelica simulation model for a product. The Implementation Tree of type geometrical can be used in the visualization of the product.

3.4 The Selection and Configuration Tool

The S*election and Configuration Tool* extends the framework by adding integrated search capabilities in FMDesign. The tool is coupled with the Implementation Tree for a Product Concept. The designer uses the selection tool to search (query) the Modelica Database for desirable simulation components to implement a certain means. An implementation object in the simulation implementation tree represents the selected simulation component. Simulation component to means mapping reflects the various design choices made by the designer. In this way, the designer can experiment with different simulation component implementations at various level of detail for a specific means. When choosing alternatives for a specific means the designer has two possibilities: to browse the repository of simulation models classified according to physical concepts or to use the search dialog. The search dialog provides the following functionality:

- Textual/pattern search of components, search for a component in a specific physical domain, search for a component with specific parameters.
- Adding/deleting a product concept specific means to simulation component mapping where the simulation component is referred from an implementation object.

After building the means-component mappings the designer can choose to edit or configure components by using the configuration dialog that provides the following functionality:

- Set implementation component parameters or parameters ranges.
- Edit the simulation component in the desired Modelica environment and use the edited component, which is also automatically added to the Modelica Database.

3.5 The Automatic Model Generator Tool

The Automatic Model Generator Tool provides the second extension of the framework.

The model generator tool has as input the Implementation Tree (Figure 3, lower right) of a product and as output the complete simulation model with the alternative design choices.

The automatic model generator traverses the Implementation Tree of a Product Concept and outputs ModelicaXML models by choosing the combination of selected components for means. The generated models are then translated to Modelica for means evaluation through simulation. To simulate the models, commercial tools like Dymola and MathModelica or the open-source OpenModelica [10] compiler can be used.

After the simulation of the generated models, the results are used as feedback for the designer. Using this feedback the designer can then choose the best-suited model, based on the simulation results.

4 Conclusions and Future Work

As future work we want to explore the use of ontologies for product concept design and for the classification of the available component libraries.

The languages developed by the Semantic Web [5] community will be used. Research efforts based on this standard are integrating experience of many promising research areas, for instance declarative rules, which still lack a vendor neutral exchange formats for industrial applications. The semantic web standard lacks important functionality for quality assurance and other necessary functionality, which today is implemented in commercial products, but will open up for sharing of important research results with industry in collaborative environments. Also we would like to improve the *Automatic Model Generator Tool* by using parts of the composition and transformation framework described in [12].

In the future we want to provide automatic evaluation through simulation of the generated models based on the constraints collected from the Product's Requirement Tree.

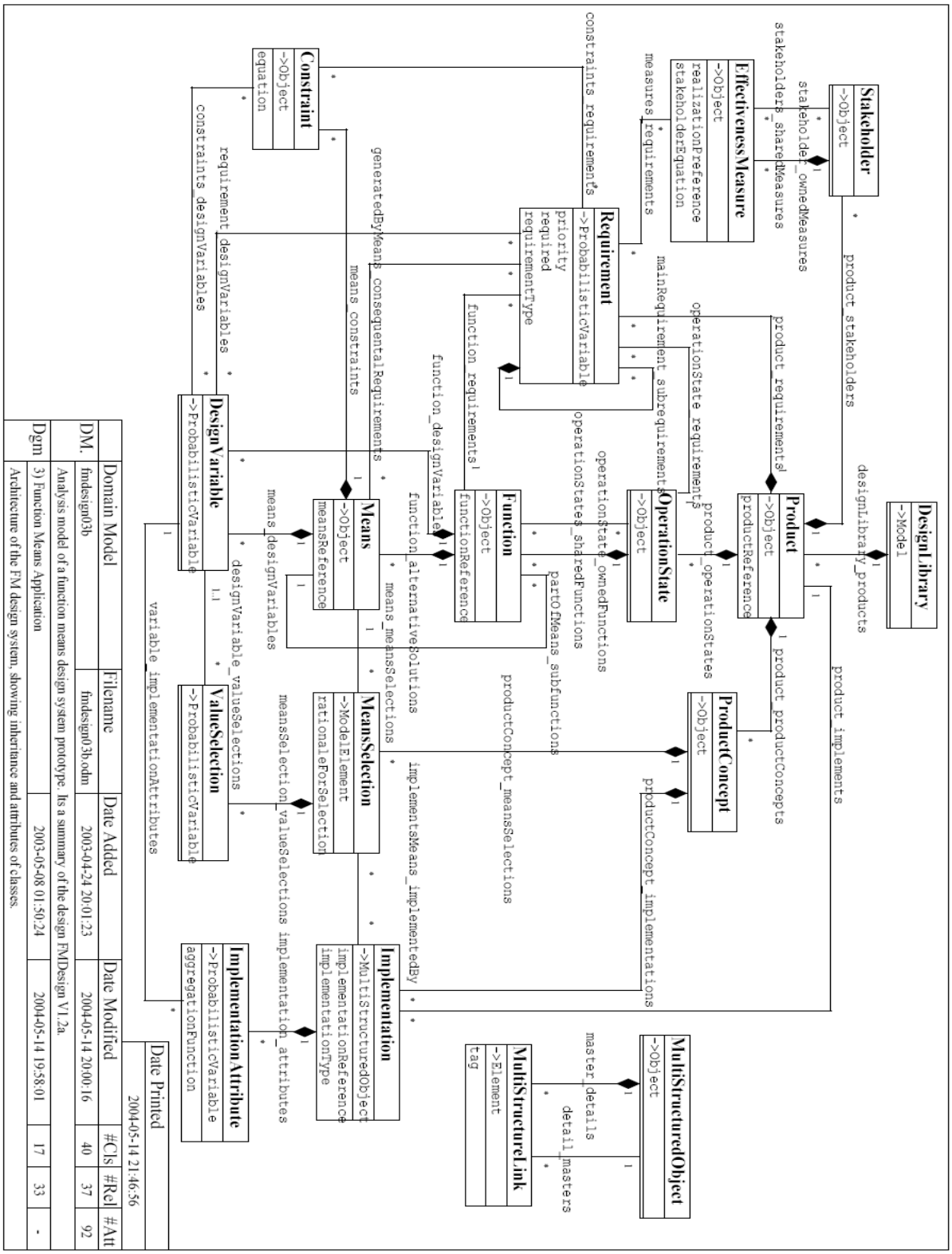
5 Acknowledgements

The ProViking research program, created by the Swedish Foundation for Strategic Research supported this research through the project *System Engineering and Computational Design (SECD)*. *The National Computer Graduate School in Computer Science (CUGS)* and *Vinnova* through the *Semantic web for products (SWEBPROD)* project.

6 References

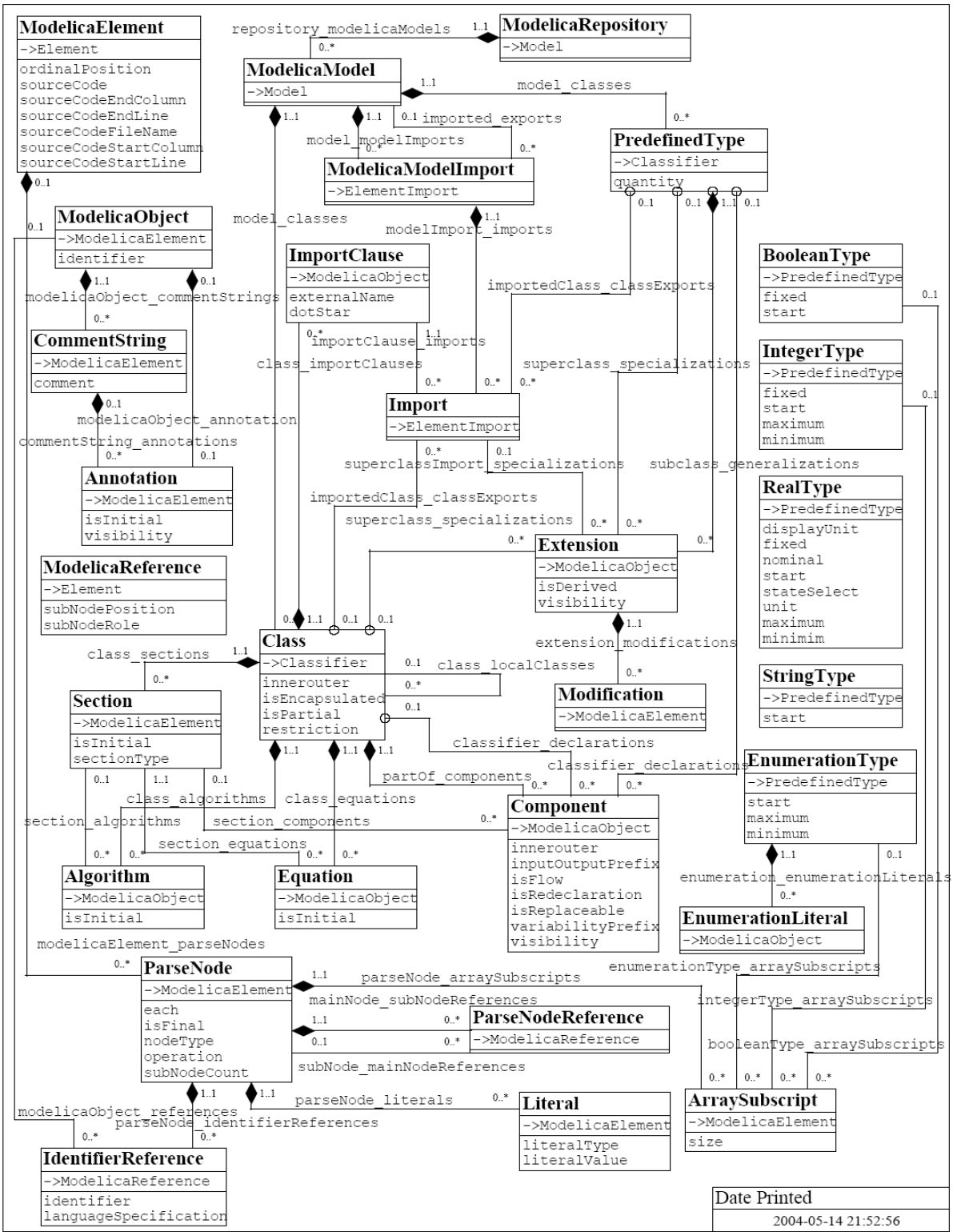
1. Dynasim. *Dymola*, http://www.dynasim.se/.
2. INCOSE. *International Council on System Engineering*, http://www.incose.org.
3. MathCore. *MathModelica*,http://www.mathcore.se/.
4. *Modelica: A Unified Object-Oriented Language for Physical Systems Modeling, Language Specification version 2.1*, Modelica Association, 2003,
5. *Semantic Web Community Portal*, http://www.semanticweb.org/.
6. OMG. *Unified Modeling Language*, http://www.omg.org/uml.
7. Mogens Myrup Andreasen. *Machine Design Methods Based on a Systematic Approach (Syntesemetoder på systemgrundlag)*, Lund Technical University, Lund, Sweden, 1980.
8. R.H. Bracewell, D.A.Bradley. *Schemebuilder, A Design Aid for Conceptual Stages of Product Design*, in *International Conference on Engineering Design, IECD'93* 1993, The Hague.
9. Peter Fritzson. *Principles of Object-Oriented Modeling and Simulation with Modelica 2.1*, Wiley-IEEE Press, 2003, http://www.mathcore.com/drmodelica.
10. Peter Fritzson, Peter Aronsson, Peter Bunus, Vadim Engelson, Levon Saldamli, Henrik Johansson and Andreas Karström. *The Open Source Modelica Project*, in *Proceedings of The 2th International Modelica Conference*,March 18-19, 2002, Munich, Germany.
11. Adrian Pop, Peter Fritzson. *ModelicaXML: A Modelica XML representation with Applications*, in *International Modelica Conference*,3-4 November, 2003, Linköping, Sweden.
12. Adrian Pop, Ilie Savga, Uwe Assmann and Peter Fritzson. *Composition of XML dialects: A ModelicaXML case study*, in *Software Composition Workshop 2004, affiliated with ETAPS 2004*,3 April, 2004, Barcelona.
13. Sören Wilhelms, *Reuse of Principle Solution Elements In Conceptual Design.* Lic. thesis 1044, Linköping University, 2003

7 Appendix



	Domain Model	Filename	Date Added	Date Modified	#Cls	#Rel	#Att
DM.	fmdesign03b	fmdesign03b.odm	2003-04-24 20:01:23	2004-05-14 20:00:16	40	37	92
	Analysis model of a function means design system prototype. Its a summary of the design FMDesign V1.2a.						
Dgm	3) Function Means Application		2003-05-08 01:50:24	2004-05-14 19:58:01	17	33	-
	Architecture of the FM design system, showing inheritance and attributes of classes.						

Date Printed: 2004-05-14 21:46:56



Date Printed
2004-05-14 21:52:56

	Domain Model	Filename	Date Added	Date Modified	#Cls	#Rel	#Att
DM.	ModelicaDB-21-02n	ModelicaDB-21-02o.odm	2003-03-04 08:03:07	2004-05-14 21:51:54	53	50	137
	Modelica 2.1 database design for ModelicaXML information exchange. The goal is a repository that can be round-tripped with Modelica source code.						
Dgm	SIMS'2004: Modelica Package		2003-03-04 09:33:10	2004-05-14 21:51:54	29	39	-
	Diagram 1: UML Model of Modelica 2.1. (Note that "->ModelicaObject", means inherits from ModelicaObject)						

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

A TOPOLOGY APPROACH TO MODELLING

Heinz A Preisig*
Department of Chemical Engineering
Norwegian University of Science and Engineering
N-7491 Trondheim, Norway
Email: Heinz.Preisig@chemeng.ntnu.no

Abstract

Going back to basic physics, combine it with discrete mathematics and system theory, resulted in a powerful tool for generating dynamic process models that are consistent with the assumptions made, consistent with the basic physical laws, where appropriate, and are guaranteed structurally solvable. The user is left the freedom to map his view of the process in the model design process, thus has all the freedom he desires, whilst being strictly watched on the basic facts of physics and system theory. The last year's experience indicates a speedup of the model building and coding process of up to a factor of 10, but quite usually 5.

Keywords: Computer-aided, modelling, DAE, process systems engineering

Nomenclature

$\underline{\mathbf{n}}$	:: Vector of molar mass [mol]
E	:: Total energy
$\underline{\hat{\mathbf{n}}}_{a\|b}$	:: Vector of molar mass flow from capacity (system) a to b [mol/s]
$\hat{q}_{a\|b}$	:: Conductive heat flow from capacity (system) a to b [mol/s]
$\underline{\hat{\mathbf{w}}}_{a\|b}$	:: Vector of work flow from capacity (system) a to b [mol/s]
$\underline{\tilde{\mathbf{n}}}_s$	:: Vector of transposition rates in capacity (system) s [mol/s]
$\underline{\mathbf{x}}_s$	:: Vector of fundamental state variables (component mass, energy) of capacity (system) s
$\underline{\mathbf{z}}_{a\|b}$	:: Vector of flows from capacity (system) a to b [mol/s]
$\underline{\mathbf{v}}_{a\|b}$	:: Vector of secondary flows from capacity (system) a to b [mol/s]
$\underline{\tilde{\mathbf{r}}}_s$	:: Vector of transposition rates in capacity (system) s in terms of extent of reactions [mol/s]
$\underline{\underline{\mathbf{F}}}$	:: Flow matrix (may be typed) [-]
$\underline{\underline{\mathbf{R}}}$	:: Transposition matrix (may be typed) [-]
$\underline{\underline{\mathbf{S}}}$	:: Selection matrix, selects from vectors elements to form a new vector [-]

For the rest of the symbols see text.

Modeller Project

Models are omnipresent in today's process systems engineering activities. Almost all methods use models in one or the other way. Thus it is not surprising that the demand for models is increasing rapidly. The Modeller project has its roots in flow sheeting that is steady state process simulators, as models for flow sheet simulators were, and to a large degree still are, essentially generated manually. Flow sheeting simulators are programs that aim at simulation the process industry's dynamic plants and construct plant training simulators.

A flow sheet is a graphical representation of a plant, which shows the different apparatuses required in the process to perform the different, individual tasks. Typical plant components are: reactors, heat exchangers, separation processes such as distillation, crystallisation, extractions, filters etc. The need for a programmed approach to constructing models for flow sheeting programs was recognised at the time the first flow sheeting programs were written in the Sixties and Seventies and several research groups have had efforts going that were aiming at generating a tool for doing so [1, 2, 4, 6, 7]. The Modeller is the result of our effort on this subject, though it is not anymore limited to flow sheeting. The modelling tool can in principle generate

*E-mail: Heinz.Preisig@chemeng.ntnu.no
web: http://www.chemeng.ntnu.no/ preisig/

models for any kind of problem, including dynamic simulation, optimisation and control design.

The Modeller has, after the third generation [3, 5, 11], reached industrial standards and is currently used as the main model building program in a company specialising on building training simulators for the chemical industry. It consists of a set of context sensitive editors with the core module being the Modeller and the other modules being used to maintain data bases on support material such as different classes of equations (transport, reaction kinetics, phase transitions, equilibrium relations, state-variable transformations and physical properties or interfaces therefore), and species, reaction information (what species and what reactions). Currently it generates output in the form of software modules that slot into the in-house solver used in the mentioned company, but also Matlab's Differential Algebraic Equation solver and Modelica.

Concepts

The Modeller is built on implementing a physical view of the world. It constructs an abstract process representation in form of a topology with two levels of refinements.

First a physical view of the space occupied by the process and its relevant environment is defined. This we call the physical topology. The first refinement can be seen as a colouring of the topology by adding the species that are present in the plant. Finally the second refinement adds the variables and equations describing the behaviour of the individual components of the topology.

Physical Topology

The first, and basic level consists of the PHYSICAL TOPOLOGY. It represents the plant as a network of PRIMITIVE CAPACITIES and CONNECTIONS. The capacities represent what one also often calls CONTROL VOLUMES, namely parts of the space to which one assigns a common property, such as some uniform intensive properties characterising a physical phase.

Generating this first level is crucial to the definition of the process model. Any following up step is limited by the structure of the physical topology. Thus it is critical to understand what it represents.

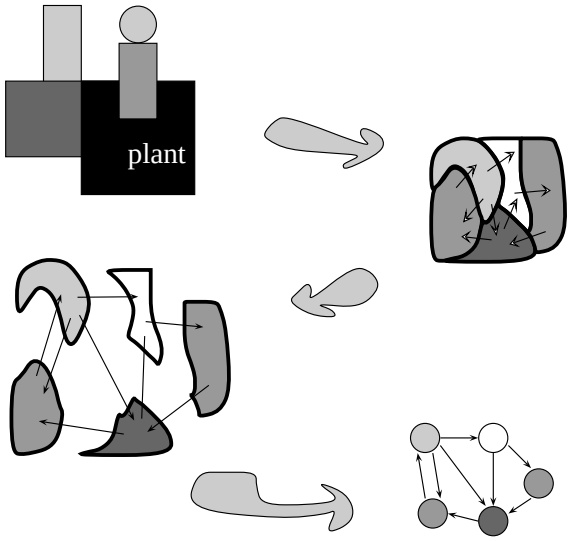


Figure 1: *Generating an abstract physical topology: split the plants volume into smaller volumes and introduce connections for extensive quantity transfer. This can be readily extended by introducing a hierarchy.*

Firstly, establishing a physical topology of a process is not an automatic process but a design process. It requires an in-depth understanding of the process being modelled. The process leading to the physical topology can be depicted as in figure 1. The user must first determine what shall be modelled, think about the environment and what affects the plant. This subdivision defines capacities and connections, thus implies a certain dynamic behaviour. This in turn links to time-scales, thus one needs to think about the time-scale range for which the model shall represent the behaviour of the real-world object. It is the WIDTH of the time-scales being modelled that implies complexity. The model will describe only those parts dynamically, which are modelled as capacities. The connections have no capacity. Relating the ability to store FUNDAMENTAL EXTENSIVE QUANTITY (= conserved quantities) with what comes in and out in terms of streams of affecting extensive quantities, gives a measure for the time scale in which the respective capacity operates. Computing the range is a mini-max calculation. The model is to match the purpose for which it is being used. Thus one has to know about the time-scale of the application of the model in order to decide how fine the GRANULARITY of the model must be. Once one has

decided on a granularity, one can refine the model by lumping too fine parts and subdivide too gross parts thus do model agglomeration or model refinement, respectively. The aim of this process must be to get a small distribution for the time constant ranges of the individual capacities.

Species Topology

The first refinement of the topology is to put the matter into the structured physical space. This is done by a couple of basic mechanisms: 1) injection of species into capacities, 2) injection of reactions into capacities, 3) constraining flows by introducing permeabilities for individual mass connections, 4) directionality of flow (bi-directional or unidirectional).

1) Injection of species: This introduces a species in a specific location. Often there is a natural point for the injection, namely the source for the species. In chemical plants this is mostly a feed tank or another resource.

2) Injection of reactions: This introduces potential reactions, that is, the reaction may take place, or one may say, the reaction is enabled. Injecting the reaction does not say anything about the conditions. It merely indicates that if a set of species is present, the reactants, than a set of other species, the products, may be generated through this transposition.

3) The permeabilities proved means to constrain flows, that is, a black-white description of a mass connection is given in which one states if or if not a species is transferred. One may think here of a absolutely selective membrane.

4) Directionality reduces the complexity, whilst constraining the descriptive power of the model significantly. In essence this last mechanism is not necessary, but reduces often the complexity. For example a feed tank that connects to a plant part such that no backwards flow can occur may be modelled in this way.

These concepts are sufficient for the computation of the species distribution. For this purpose, one makes use of the typing of connections as one looks only at the connections that can transfer species, mass connections. The typing is usually done when defining the physical topology, but it may also be done later. The typing allows also a distinction of topologies. One can view coloured, or typed, networks and define for example mass transfer networks related to mass in general or species mass or any combination of species as a colour. One may do the same for other quantities such as heat or work, etc. The computation of the species distribution is then simply an extended colouring algorithm as one can find them in any discrete mathematics text that talks about graphs and their properties etc.

Variable and Equation Topology

Finally to each element in the topology a mathematical description is added. The basic dynamic description is given by a basic dynamic description of the individual capacities and the physical topology combined with the species topology. For each capacity a set of component mass balances and an energy balance as well as momentum balances can be generated when introducing the concept of INDUCED FLOWS. Latter are easiest explained on examples, namely mass flow induces energy flow as mass carries internal energy, potential energy and kinetic energy. Equally, mass flow induces momentum flow. Assigning symbols to molar component mass, $\underline{\mathbf{n}}$, total energy, E, conductive and radiation heat flow, $\hat{q}$, work flows, $\hat{w}$, one can write the component mass balances and the energy balance [11]. For the two

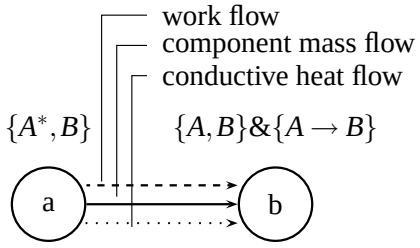


Figure 2: *Two simple systems communicating component mass, work and conductive heat. Species A was injected in system a (marked with a * in the species set), is transferred to system b through the mass connection. In system b there is a possible reaction of $A \rightarrow B$. Thus species B is generated and again transferred back to system a through the mass connection. In addition work and convective heat is communicated through the respective connections.*

systems in Figure 2 one can write the component

mass balances:

$$\frac{d\underline{\mathbf{n}}_a}{dt} := -\underline{\hat{\mathbf{n}}}_{a|b}\,, \quad (1)$$

$$\frac{d\underline{\mathbf{n}}_b}{dt} := \underline{\hat{\mathbf{n}}}_{a|b} + \underline{\tilde{\mathbf{n}}}_b\,. \quad (2)$$

With the hats indicating flows and the notation $a|b$ giving the direction of the flow, namely from system a to system b. The direction indicated is to be seen as reference coordinate system against which the flow of the individual components are measured. The same applies to all the other quantities. The graphs are thus directed. For the energy balances we can write:

$$\frac{dE_a}{dt} := -\hat{E}_{a|b} - \hat{q}_{a|b} - \hat{w}_{a|b} - \hat{w}(\underline{\hat{\mathbf{n}}}_{a|b})\,, \quad (3)$$

$$\frac{dE_b}{dt} := \hat{E}_{a|b} + \hat{q}_{a|b} + \hat{w}_{a|b} + \hat{w}(\underline{\hat{\mathbf{n}}}_{a|b})\,. \quad (4)$$

Notice that the energy balances are given in their pure form, thus no transformation of any kind has been applied. Also, there are induced flows to be observed, here volume work flows $\hat{w}(\underline{\hat{\mathbf{n}}}_{a|b})$ that are induced by mass flow besides the already mentioned energy flow, $\hat{E}_{a|b}$. The energy E is including internal, kinetic and potential energy. Because no state variable transformations have been yet applied, the reaction term is not appearing in any form in the energy balance of system b. It should also be noted that the directionality reflects into the balances and it is easy to see that one can map all balances into a block matrix equation:

$$\frac{d\underline{\mathbf{x}}}{dt} := \underline{\underline{\mathbf{F}}}\,\underline{\mathbf{z}} + \underline{\underline{\mathbf{R}}}\,\underline{\mathbf{r}}\,. \quad (5)$$

Where $\underline{\mathbf{x}}$ is the vector of fundamental quantities, a stack of the component mass and energy for the two systems. The vector $\underline{\mathbf{z}}$ is the stack of flows and the vector $\underline{\mathbf{r}}$ the vector of transpositions, here reactions. The matrix $\underline{\underline{\mathbf{F}}}$ is the flow matrix and contains only information about the graph thus blocks of -1 and $+1$, whilst the matrix $\underline{\underline{\mathbf{R}}}$ is a block matrix with the blocks being the stoichiometric matrices for the respective systems.

The flows and the reactions introduce secondary state variables such as concentrations and the POTENTIALS temperature, pressure and chemical potential. These secondary state variables MUST be the result of a mapping from the primary, fundamental state $\underline{\mathbf{x}}$. Whilst this seems obvious, the MODELLER program, which was the result of three consecutive PhD thesises [3, 5, 11], is the first program that actually enforces these mappings. This has been introduced in the third thesis by Westerweele.

The algebraic equations listed below must be assigned to the flows and reactions:

$$\underline{\mathbf{z}} := \left[\underline{\underline{\mathbf{S}}}_z, \underline{\underline{\mathbf{S}}}_v\right] \begin{bmatrix} \underline{\mathbf{z}}(\underline{\mathbf{y}}, \underline{\mathbf{p}}_z) \\ \underline{\underline{\mathbf{S}}}_{yv}\,\underline{\mathbf{y}}\;\underline{\mathbf{v}}(\underline{\mathbf{y}}, \underline{\mathbf{p}}_z) \end{bmatrix}, \quad (6)$$

$$\underline{\mathbf{r}} := \underline{\mathbf{r}}(\underline{\mathbf{y}}, \underline{\mathbf{p}}_r)\,, \quad (7)$$

Finally the secondary states and the properties are extended.

$$\underline{\mathbf{y}} := \underline{\mathbf{y}}(\underline{\mathbf{y}}, \underline{\mathbf{x}}, \underline{\mathbf{p}}_y)\,, \quad (8)$$

$$\underline{\mathbf{p}}_j := \underline{\mathbf{p}}_j(\underline{\mathbf{y}}, \underline{\mathbf{p}}_p) \quad ; j \in \{z, r, y, p, i\}\,, \quad (9)$$

The resulting equations must generate a completely defined set otherwise the system is not proper [9]. Whilst some of these equations are not explicit, it is in most cases possible to arrange them in a lower triagonal form. The analysis can be facilitated by a bi-partite graph analysis. The concept of LOCALITY is essential: Connections are local to the two connected systems, whilst the rest of the algebraic part of the model is local to the individual system. The analysis is thus limited always to single or pairs of systems. In the case of our implementation in MODELLER, this analysis is done on-line as the equations are being selected from a pre-defined set of lists. The relative complex representation of the flow equations allows for the use of transformations first. It is quite common that, for example, mass flow is given in terms of volumetric flow.

Time-scale assumptions

The ability to make time-scale assumptions in the form of fast transfer, fast kinetics and small capacities (all reaching the respective limit of infinity or zero) and resolve the resulting structural problems in the differential algebraic model is unique to the current implementation of the MODELLER. The index problem is resolved through model reduction, which utilises the fact that the balance equations are linear. The model reduction reduces consequently to null-space computations of submatrices of the flow matrix and the transposition matrix. For details the reader is referred to [8, 10, 11].

Implementation

The current implementation of the MODELLER has the following functions:

- Context sensitive editor for manipulating a hierarchical physical topology
- Refinement with species topology
- Refinement with variable and equation topology
- On-line check on consistency of equation set
- Definition of simulation models
- On-line implementation of time-scale assumptions with automatic model reduction to index 1 problems
- Generation of code for matlab's DAE solver (MathWorks), e-modeller (Protomation BV, NL), Modellica (Dynasim).

Besides the brief example to follow, the interested reader is referred to the thesis of Westerweele [11], which can be found on the web page of the author. The current implementation is limited to lumped systems, component mass and enthalpy, thus constant pressure system. A project to extend into distributed systems is currently on its way. Further, efforts are taken to make the input to the equation topology harder, that is, a context sensitive editor checking on the validity of thermodynamic relations is now defined as a new project.

A Brief Example

For the purpose of demonstrating some of the features of the discussed representation, we look at the equations for a very simple plant Figure 3. The first step is to suggest a physical topology. We use a simple concept by assuming the behaviour of an ideally stirred tank reactor for the jacket (J) and the contents of the reactor (F). The feed and product tanks as well as the source and sink of the two streams serving the jacket, are not of interest and are modelled as infinitely large capacities, thus thermodynamic reservoirs (Figure 4). Also, we limit the representation to component mass and energy. Notice that the figure indicates a two-level hierarchy in the

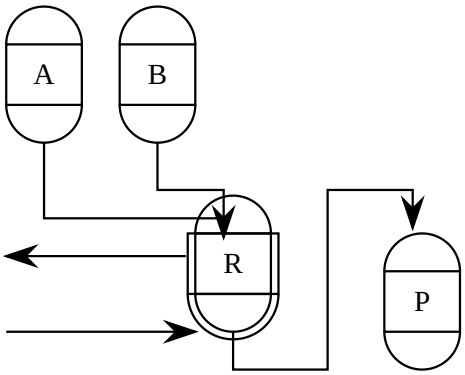


Figure 3: *A simple reactor installation with two feed tanks, a jacketed stirred tank reactor and a product tank*

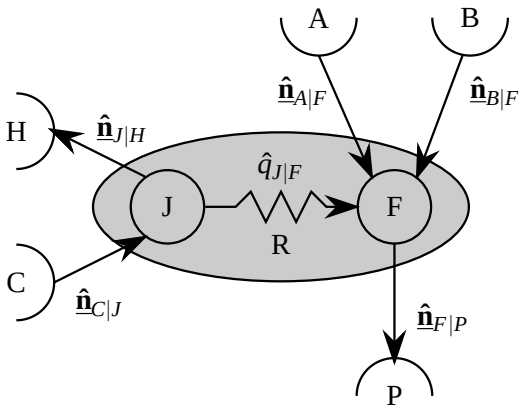


Figure 4: *Assuming an ideally stirred tank behaviour for the reactor tank contents and the jacket contents, and only being interested in the reactor and not the feed as well as the product tank, the topology is rather simple.*

representation, as the reactor is shown as a subnetwork. In a next step, we introduce the "chemistry" adding the species A and S to tank (reservoir) A, B and S to tank (reservoir) B and the cooling fluid K to reservoir C. In the fluid phase of the reactor we introduce a chemical reaction in which A and B are reacting to form species D. The topology can be typed generating coloured topologies. If we choose to show the mass domains we only have to delete all the heat-flow connections and, if they would be present, the work-flow connections, to find two of them (Figure 5). For the heat-flow domains we delete all mass flows and, if they would be present, the remaining work flows (Figure 6). With this simple structure, we only have to generate the equations for the two lumps: jacket and fluid con-

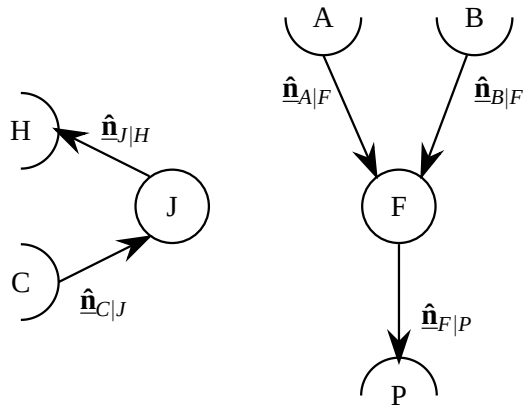


Figure 5: *The two mass-flow domains give the mass-flow matrix* $\underline{\underline{\mathbf{F}}}^m$

Figure 6: *The heat-flow domains with the reservoirs being primitive domains. This graph yields the convective heat-flow matrix* $\underline{\underline{\mathbf{F}}}^q$

tents. Before we do that so, we introduce a set of early assumptions, which are quite commonly, not to say nearly always, made in such systems. The assumptions are related to the energy balance. Let us write a generic energy balance for an arbitrary network:

$$\begin{aligned}\frac{d\underline{\mathbf{E}}}{dt} &= \underline{\underline{\mathbf{F}}}^m\,\underline{\hat{\mathbf{E}}} + \underline{\underline{\mathbf{F}}}^q\,\underline{\hat{\mathbf{q}}} + \underline{\underline{\mathbf{F}}}^m\,\underline{\hat{\mathbf{w}}}(\underline{\hat{\mathbf{n}}}) + \underline{\underline{\mathbf{F}}}^w\,\underline{\hat{\mathbf{w}}},\\ &= \underline{\underline{\mathbf{F}}}^m\,\left(\underline{\hat{\mathbf{E}}} + \underline{\hat{\mathbf{w}}}(\underline{\hat{\mathbf{n}}})\right) + \underline{\underline{\mathbf{F}}}^q\,\underline{\hat{\mathbf{q}}} + \underline{\underline{\mathbf{F}}}^w\,\underline{\hat{\mathbf{w}}}.\end{aligned}$$

We see clearly the effect of defining induced flows, here volumetric work flow, which is induced by the mass flow. The total energy (E) is the sum of internal (U), kinetic (K) and potential (P) energy:

$$E \;:=\; U+K+P\,.$$

The effect of kinetic and potential energy, in the capacities as well as in the mass-flow streams, are negligible which gives rise to the definition of enthalpy

$$H \;:=\; U+pV\,,$$

which leads to a simpler representation

$$\frac{d\underline{\mathbf{H}}}{dt} \;=\; \underline{\underline{\mathbf{F}}}^m\,\underline{\hat{\mathbf{H}}} + \underline{\underline{\mathbf{F}}}^q\,\underline{\hat{\mathbf{q}}} + \underline{\underline{\mathbf{F}}}^w\,\underline{\hat{\mathbf{w}}}.$$

Finally, before we can generate the equations we need to determine the component-mass-flow domains. These depend on the assumptions of bi-directional or uni-directional mass flows. In the case we have here, it would be natural to assume uni-directional flow, meaning that one a priori eliminates the possibility that the mass stream flows in negative direction, that is, opposite the arrows' direction. The difference is seen quickly:

cap.	reac.	uni-dir.	bi-dir.
A		$\{A^*,S\}$	$\{A^*,B,D,S\}$
B		$\{B^*,S\}$	$\{B^*,B,D,S\}$
F	$\{A+B\rightarrow D\}$	$\{A,B,D,S\}$	$\{A,B,D,S\}$
P		$\{A,B,D,S\}$	$\{A,B,D,S\}$
J		$\{K\}$	$\{K\}$

Now it is straightforward to generate the component mass balances and the energy balances for the two capacities J and F.

$$\begin{bmatrix}\frac{d\underline{\mathbf{n}}_F}{dt}\\ \hline\hline \frac{d\underline{\mathbf{n}}_J}{dt}\end{bmatrix} \;:=\; \underline{\underline{\mathbf{F}}}^{\mathbf{m}}\begin{bmatrix}\underline{\hat{\mathbf{n}}}_{A|F}\\ \underline{\hat{\mathbf{n}}}_{B|F}\\ \underline{\hat{\mathbf{n}}}_{F|P}\\ \hline\hline \underline{\hat{\mathbf{n}}}_{C|J}\\ \underline{\hat{\mathbf{n}}}_{J|H}\end{bmatrix} + \underline{\underline{\mathbf{R}}}\,\underline{\tilde{\mathbf{n}}}_F\,,$$

For uni-directional flows, the mass-flow matrix $\underline{\underline{\mathbf{F}}}^{\mathbf{m}}$ is:

$$\left[\begin{array}{ccc|ccc||cc}
1 & & & -1 & & & & \\
 & 1 & & & -1 & & & \\
 & 1 & 1 & & & -1 & & \\
\hline\hline
 & & & & & & 1 & -1
\end{array}\right].$$

The two mass domains stand out clearly as the two blocks (lower left, upper right) are zero. Also notice the compactness of the representation. There is no unnecessary information included, for example only those species are being included that are actually present. There are also no "cut-equations" as they are often used in simulation software such as Modelica and bondgraph programs. The stoichiometric matrix is also easily found

$$\underline{\underline{\mathbf{R}}} \;:=\; \begin{bmatrix}-1\\ -1\\ 1\\ 0\\ \hline\hline 0\end{bmatrix}.$$

Finally one would define the expressions for the uni-directional mass flows (select second option in equa-

tion 6) and the 2nd order reaction kinetics:

$$\begin{aligned} \underline{\mathbf{n}}_{a|b} &:= \underline{\mathbf{c}}_a \hat{V}_{a|b}\,, \\ \underline{\mathbf{r}}_F &:= k(T_F)\, c_{F,A}\, c_{F,B}\,. \end{aligned}$$

The kinetic constant could be a function of the temperature as indicated, which usually is modelled with an Arrhenius equation. Both these define concentrations, which are secondary state variables that need to be linked back to the primary state. For the arbitrary system a these are the equations:

$$\begin{aligned} \underline{\mathbf{c}}_a &:= \frac{n_a}{V_a}\,, \\ V_a &:= [1,...,1]\,\underline{\mathbf{n}}_a\, \rho_a^{-1}\,. \end{aligned}$$

with ρ being the molar density. The energy balances are:

$$\begin{bmatrix} E_F \\ E_J \end{bmatrix} = \underline{\underline{\mathbf{F}}}^m \begin{bmatrix} \hat{H}_{A|F} \\ \hline \hat{H}_{B|F} \\ \hline\hline \hat{H}_{F|P} \\ \hline\hline \hat{H}_{C|J} \\ \hline \hat{H}_{J|H} \end{bmatrix} + \underline{\underline{\mathbf{F}}}^q\, \hat{q}_{J|F}\,.$$

The mass flow matrix does here only refer to the total flows (thus the modified symbol $\underline{\underline{\mathbf{F}}}^m$ instead of $\underline{\underline{\mathbf{F}}}^{\underline{\mathbf{m}}}$):

$$\underline{\underline{\mathbf{F}}}^m := \left[\begin{array}{c|c|c||c|c} 1 & 1 & -1 & 0 & 0 \\ \hline\hline 0 & 0 & 0 & 1 & -1 \end{array}\right].$$

The conductive heat-flow matrix is:

$$\underline{\underline{\mathbf{F}}}^q := \left[\begin{array}{c} 1 \\ \hline\hline 0 \end{array}\right].$$

The heat transfer is modelled with Newton's law of cooling:

$$\hat{q}_{J|F} := -p_{J|F}\,(T_F - T_J)\,.$$

The temperature is to be computed from the relation:

$$H := \int_{T_{\text{ref}}}^{T} c_p(T)\, dt\,.$$

Where $c_p(T)$ is the specific heat capacity, which is the partial derivative of the enthalpy with respect to the temperature and the "parameter" $p_{J|F}$ is the product of the overall heat transfer coefficient and the heat transfer area.

These equations give the main relations. The little being left out should be easy to fill in by the reader and complete the description. The attentive reader will also notice that the equation set defines a bi-partite graph, which can be used to establish what must be given in order to be able to integrate the equations. In the view of the limited space this is left to the reader, but given the initial conditions for the primary state variables being the molar masses in the capacities J and F and the parameters (kinetic constant, density etc.), a DAE solver will be readily able to solve these equations.

A Thought on the Side

If one looks into the current contents of education programs in process engineering, than one finds that there is very little education on the structuring mechanism as it underlies this analysis. In most cases it is hidden away or in the best cases hinted such as "making a pseudo-steady state assumption". It is astonishing that whilst any model is requiring going through this process, the required thinking pattern is currently not addressed explicitly in most of our teaching programs.

Also: whilst electrical and mechanical engineers typically deal with scalar or 3-dimensional spaces for the primitive model components, the chemical engineers' primitive elements are usually of higher dimension, namely of # of mass components + 1 for energy + possibly 3 for momentum. Thus it is surprising that it is chemical engineering that has the smallest amount of education in multidimensional spaces when comparing the three disciplines.

Conclusions

Models are not unique items. For every plant one may design different models. It is the user, who is asked to provide his view of how the physical space of the plant and its affecting environment by splitting the identified domain into a set of primitive systems that have capacity to store mass, energy etc, and connections that communicate extensive quantities between pairs of neighbouring primitive systems. The result is an abstraction of the control volume concept into a graph with the vertices representing the capacities and the arcs representing the connections.

Adding the colouring, where colours are species and type of extensive quantity, the main body of information about the process model is captured, namely the model granularity, the interactions and the relevant extensive quantities. After having defined the granularity and interaction pattern of the physical - chemical - biological entity, one only needs to fill in the mechanisms of (i) transfer of extensive quantity, (ii) chemical (biological) kinetic and phase transitions, (iii) state variable transformations and (iv) physical properties all as a function of the fundamental state, latter being the vector of conserved quantities. The graph, thus, contains all this latter information and is, as a picture, very well suited for communicating the process model properties in the large. We have not only been using this concept for the Modeller, but use it also for teaching and any kind of discussion, where the properties of a model are of relevance. Today, models are build with little analysis as indicated above and probably are linking to the main problems in modelling dynamic processes.

The MODELLER is the first program of its kind that guarantees the generation of STRUCTURALLY SOLVABLE SIMULATION PROBLEMS, namely DIFFERENTIAL ALGEBRAIC EQUATIONS OF INDEX 1

The key to this achievement is to not substitute the algebraic part but treat the problem in the fundamental space of the conserved quantities, which is linear in the essential quantities.

The project has reached industrial standard by proving its efficiency in the construction of training simulators and other simulation models. A 10-fold increase in efficiency is not a-normal, but 3-5 times quicker is almost always achieved. Whilst the tool is still somewhat experimental and suffers of some shortcomings, these problems will sequentially be removed in the follow-up projects.

References

[1] J Bieszczad. *A framework for the language and logic of computer-aided phenomena-based process modeling*. PhD thesis, 2000.

[2] J A Krogh. *Generation of problem specific simulation models with an integrated computer aided system*. PhD thesis, Technical University of Denmark, Kastrup, Denmark, 1998.

[3] T Y Lee. *The Development of an Object-Oriented Environment for the Modelling of Physical, Chemical and Biological Systems*. PhD thesis, Texas A & M University, College Station, TX, USA, 1991.

[4] W Marquardt. Dynamic process simulation–recent progress and future challenges. In *Dynamic Process Simulation–Recent Progress and Future Challenges*, New York, 1992. CPC IV, CACHE-AIChE Publications.

[5] A Z Mehrabani. *Computer aided modelling of physical-cyhemical-biological systems*. PhD thesis, University of New South Wales, Sydney, Australia, 1995.

[6] H I Moe. *Dynamic process simulation: studies on modeling and indes reduction*. PhD thesis, Norwegian University of Science and Technology, Trondheim, Norway, 1995.

[7] J D Perkins, R W H Sargent, R Vasquez-Roman, and J H Cho. Computer generation of process models. *Computers & Chemical Engineering*, 20:.635–639, 1996.

[8] H A Preisig. On concentration control of fast reactions in slowly-mixed plants with slow inputs. pages FP 06–6, 2002.

[9] H A Preisig. Modelling: Compartmental networks and topologies. In *Modelling: Compartmental Networks and Topologies*. ESCAPE 14, 2004.

[10] H A Preisig and M R Westerweele. Effect of time-scale assumptions on process models and their reconciliation. In *Effect of Time-Scale Assumptions on Process Models and Their Reconciliation*. ESCAPE 13, 2003.

[11] M R Westerweele. *Five Steps for Building Consistent Dynamic Process Models and Their Implementation in the Computer Tool MODELLER*. PhD thesis, TU Eindhoven, Eindhoven, The Netherlands, ISBN 90-386-2964-8 2003.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

FIELD TYPE AND FIELD CONSTRUCTOR IN MODELICA

Levon Saldamli and Peter Fritzson
Dept. of Computer and Information Science
Linköpings universitet
SE-581 83 Linköping, Sweden
{levsa,petfr}@ida.liu.se

Abstract

Modelica is an equation-based, mathematical modeling language which is in the process of being extended with support for partial differential equations (PDEs). In this context, support for distributed variables, so called fields, is one of the most important extensions. Fields will be defined over continuous domains in the language, in order to separate mathematical description from implementation details like discretization. In this paper, we describe declaration of fields over continuous domains, initialization of field values, mapping between fields and scalars and vice versa, and also type compatibility in expressions containing fields.

Keywords: Component-based modeling. Object-oriented modeling. Declarative modeling. Partial differential equations.

Introduction

Modelica is a general purpose, object oriented modeling and simulation language [7, 11]. Physical systems can be modeled declaratively in Modelica using equations, and complex systems can be defined hierarchically using components and connections. Physical properties are declared as time-dependent variables and equations are written using expressions containing variables and time derivatives of variables. Before simulation, the models are translated by a Modelica compiler into a system of differential and algebraic equations (DAEs), which can be solved using standard DAE solvers. Integrated Modelica tools such as Dymola [4] and MathModelica [6, 9] exist, where translation and simulation is done automatically by the tool.

An example model of a capacitor follows:

```
model Capacitor
  import Modelica.SI.*;
  parameter Capacitance C=1;
  Voltage v;
  Current i;
equation
  i = C*der(v);
end Capacitor;
```

Here, `Capacitance`, `Voltage` and `Current` are types declared in the Modelica Standard Library [10] and are derived from the builtin basic type `Real` representing real values. The variable `C` is declared as a parameter, which means that its value is constant during simulation. The variables `v` and `i` are time-dependent variables. The equation states the relationship between the current and the voltage that holds for a capacitor. Depending on the circuit that this model is used as a component, the Modelica tool automatically solves for the unknown variable and represents the results. The parameter can be modified during declaration of capacitor components, or at the start of a simulation.

Currently, Modelica has very limited support for modeling with partial differential equations (PDEs). There is ongoing research for introducing language constructs for direct PDE support in the Modelica language [16].

This paper describes language constructs to support time-dependent variables with spatial distribution. The idea is to define a representation of the `field` concept in mathematics, with continuous ranges instead of arrays as in common programming languages, in order to allow mathematical models to be independent of specific implementation choices. Language constructs are needed for definition of continuous ranges, declarations of fields on domains

and definition of field values using a construct similar to array constructors with iterators in Modelica and other languages such as Python [15]. Also, operators for converting spatially distributed variables into scalar variables and distribution of scalar variables on a domain are needed, in order to connect distributed models and models with scalar variables together. The implementation of the extensions in a prototype environment called OpenModelica [1, 13] is also described, together with an example.

In the following sections, the syntax of language constructs is shown using the following notation:

"**parameter**" *Type var* ";"

In this notation, the characters between quotes are taken literally. Words in bold style denote keywords, and emphasized words denote variable grammar elements.

Fields

Quantities with spatial distribution are usually represented using arrays in programming languages as well as in modeling and simulation languages such as Modelica. In such cases, the variable is discretized during formulation of the model. In order to declare the model before doing any discretization, a new kind of type is needed in Modelica, called a *field* type. A field is a mapping from points in a domain to values. In mathematical software the domain is usually discretized and the field is an array of values for each of the discrete points. In order to separate discretization of a model from the model declaration, fields in Modelica should be defined over continuous domains.

Field Variables

Type specification in variable declarations in Modelica can contain attributes such as `parameter`, `flow`, etc., as seen in the example in the introduction. The type attributes are used to declare different kinds of variables with respect to variability, signal direction, etc. Using a keyword *field* to specify fields allows declaration of fields with any existing type. The syntax is as follows:

"**field**" *Type fieldvar* ";"

Here, *Type* can be any of the builtin types or user-defined types derived from the builtin types. For example:

```
field Temperature T;
```

`Temperature` can be defined as follows:

```
type Temperature = Real(unit="K");
```

When declaring a field variable, the definition domain must be specified as well. In Modelica, builtin types have attributes such as `unit`, `start`, etc., which can be modified in declarations or new type definitions. A new builtin attribute *domain* in the `Real` type can be used to give the definition domain of a field variable using modifications. Modifications in Modelica are hierarchical parenthesized lists of name=value equations in declarations. For instance, a modification is used in the definition of the `Temperature` type above, for changing the `unit` attribute of the builtin type `Real` to the unit value that holds for temperature. The declaration of `T` above with a domain modification looks like this:

```
RectangularDomain omega(...);
field Temperature T(domain=omega);
```

Here, a domain of type `RectangularDomain` is used. Domain definitions in Modelica are further described in [16].

Field Constructor

In order to initialize field variables or define constant fields, constant field values must be possible to express in the language. A field constructor is an expression which generates a field value over a given domain using a given expression. The syntax for the field constructor is:

"**field**" "**(**" *expr1* "**for**" *iter* "**in**" *expr2* "**)**"

Here, *iter* is a single iterator variable or a list of iterator variables enclosed in parenthesis. The iterator variables represent the coordinates in the space where the domain is defined. For example, for a two-dimensional domain, two iterator variables are needed, e.g.:

```
TestDomain omega(...);
field Real f = field(2*x+y for (x,y) in omega);
```

The iterator variables are defined only in the local scope, e.g. they are only available in the field constructor expression. Variables in the outer scope

with the same names as the iterator variables are shadowed.

The first expression *expr1* is the expression for calculating the field values. Given a point in the domain, the value of the field at that point is the value of *expr1*. The iterator variables can be used in *expr1* to define fields with spatial dependency.

The second expression *expr2* should evaluate to a domain object. Currently, this can only be a reference to a previously declared domain object. In future implementations, domain constructor expressions may be considered as well.

This approach is similar to the existing array constructor syntax in the current Modelica language, where the keyword `array` is used instead of `field`. Array constructors in Modelica also have a shorter syntactic form:

```
"{" expr1 "for" iter "in" expr2 "}"
```

A corresponding short form for field constructors was considered, but using curly braces in both cases makes the semantics ambiguous and requires type checking of *expr2* to determine whether the expression is an array or a field constructor. Other characters than curly braces might be considered for the short form of the field constructor.

Field Type in Expressions

When field variables appear in expressions, the type of the expressions must be derived appropriately. Mixing of field variables and scalar variables must be handled, as well as field variables with different definition domains.

Binary Operators

For mixing of scalars and fields, some of the binary operators can be overloaded in a similar way as for vectors and matrices in Modelica. Table 1 shows the binary operators in Modelica which are overloaded to handle operations between scalars and arrays. Similar overloading can be done for operations between scalars and fields.

The elementwise array operations require the sizes and dimensions of the operand arrays to be equal. Correspondingly, elementwise field operations should require the domains of the operands to be equal. Table 2 summarizes some binary operators and their result types when used with fields or scalars and fields.

Table 1: Array operations in Modelica

	Operand 1	**Operand 2**	**Result**
=, :=	Array[m,n]	Array[m,n]	Array[m,n]
+, -	Array[m,n]	Array[m,n]	Array[m,n]
*****	Scalar	Array[m,n]	Array[m,n]
	Array[m,n]	Scalar	Array[m,n]
	Array[m,n]	Array[n,p]	Array[m,p]
/	Scalar	Array[m,n]	Array[m,n]
	Array[m,n]	Scalar	Array[m,n]

Table 2: Overloading of binary operators for fields

	Operand 1	**Operand 2**	**Result**
=, :=	Scalar	Field (on Ω)	Field (on Ω)
	Field (on Ω)	Scalar	Field (on Ω)
	Field (on Ω)	Field (on Ω)	Field (on Ω)
+, -	Scalar	Field (on Ω)	Field (on Ω)
	Field (on Ω)	Scalar	Field (on Ω)
	Field (on Ω)	Field (on Ω)	Field (on Ω)
***, /**	Scalar	Field (on Ω)	Field (on Ω)
	Field (on Ω)	Scalar	Field (on Ω)
	Field (on Ω)	Field (on Ω)	Field (on Ω)

Although addition and subtraction of scalars and arrays is currently not directly supported by overloaded operations in Modelica, this can easily be arranged for fields and scalars. Addition or subtraction of a scalar with a field can be seen as changing the value of the field with a constant offset.

The equation and assignment operators will also have an optional domain attribute, which is useful for defining boundary conditions. The syntax for equation and assignment with an explicit domain is:

```
expr1 "=" expr2 "in" expr3;
cref ":=" expr2 "in" expr3;
```

Here, *expr3* is a domain object, which can currently only be a reference to a previously declared domain. An equation with a domain attribute can look like this:

```
model Test
  TestDomain omega(...);
  field Real f(domain=omega);
  parameter Real q1,q2;
equation
  f = q1+q2  in omega.boundary;
end Test;
```

The fields appearing in the expression should still have similar definition domains. The explicit domain attribute should then be a subset of the domain of the fields appearing in the expression. Possibly, this requirement can be relaxed, to only requiring that the explicit domain must be a subset of all appearing fields, which may have different domains with at least the same subset as the explicit domain.

An optional extension is to also allow the keyword `on` for use with domain objects that represent boundaries, as it is used in mathematical literature when referring to boundaries e.g. in boundary conditions.

Special Operators

The builtin special operators in Modelica should be overloaded to handle fields as well. For example, the **der**() operator applied to v, i.e. **der**(v), represents the derivative of the variable v. If v is a field, the result of the operation should be a field with the same domain as v and where the resulting field is the derivative of the field v. The same applies to the new operator **pder**() which represents a partial derivative. Special operators which take more than one argument, like **pder**(), should check for the first argument for type information, and keep the domain of the given field variable.

For example, the operator **der**(v) can be defined for a field argument on a one dimensional domain using a field constructor:

```
der(v) = field(der(v(x)) for x in v.domain);
```

Here, v(x) is a field access operation with a scalar variable, as described in the following section, and **der**(v(x)) is the time derivative of the scalar variable according to standard Modelica.

Accessing Field Values

Values of the field on singular points in the domain can be accessed using the function-call like syntax:

```
fieldvar "(" expr [ , expr ...]  ")"
```

Here, *fieldvar* is a reference to a field variable, and the list of expressions is the coordinates in the domain of the field variable. The number of expressions given here should be equal to the domain dimension. For example:

```
model TempField
  field Temperature T(domain=omega);
  Temperature t_interior;
equation
  t_interior = T(1.4, 1.7);
end TempField;
```

Here `omega` is a two dimensional domain, and the equation states that the scalar variable `t_interior` is equal to the value of the field `T` at the coordinates $(1.4, 1.7)$ in the domain.

Field reduction

Integrals appear in equations in many modeling situations. In order to express integrals of fields over spatial variables, an integral operator is needed. This can be seen as a field reduction operator. Reduction expressions for arrays are already supported for some operations in Modelica. Hence, the syntax for the integral operator should be similar to the existing syntax:

```
"integral" "(" expr1 "for" iter "in" expr2 ")"
```

Here, *iter* is a list of local variables representing coordinates in the domain given in *expr2*. The result of the integral operation is a scalar. Optionally, single integration can be supported, i.e. integrating over fewer spatial variables than the dimension of the domain, resulting in a field with a smaller dimension.

Example

A two-dimensional Modelica example where fields are used is shown in Figure 3. The domain of the problem, shown in Figure 3, consists of two subdomains, `plate` and `heater`. The partial differential equation in the plate part is

$$\rho C \frac{\partial T}{\partial t} - \nabla \cdot (k \nabla T) = 0 \qquad \text{in} \quad \Omega_{plate} \qquad (1)$$

Here, ρ and C are the material constants density and heat capacity, k is the thermal conductivity, and T is the unknown temperature. In the heater part, the source term Q_h is non-zero:

$$\rho_h C_h \frac{\partial T}{\partial t} - \nabla \cdot (k_h \nabla T) = Q_h \qquad \text{in} \quad \Omega_{heater} \qquad (2)$$

The homogeneous Neumann boundary condition is used for the insulated edges, stating that no heat flow occurs through that edge:

$$k \frac{\partial T}{\partial n} = 0 \qquad \text{on} \quad \Omega_{ins} \qquad (3)$$

For the non-insulated side, the Robin boundary condition is used, where heat flow through the boundary is proportional to the temperature difference across the boundary:

$$k_n \frac{\partial T}{\partial n} + h_n T = h_n T_{out} \qquad \text{on} \quad \Omega_{nonins} \qquad (4)$$

Here, k_n is the thermal conductivity of the boundary, h_n is the heat transfer coefficient of the boundary and T_{out} is the temperature outside the domain. Both boundary conditions (3) and (4) use the normal derivative $\frac{\partial}{\partial n}$, which is the directional derivative in the outward normal direction from the boundary. In the code shown in Figure 3, the normal derivative is represented by the operator **nder**().

In two dimensions, the differential operators divergence and gradient are defined as:

$$\textit{Divergence}: \quad \nabla \cdot \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \frac{\partial u_1}{\partial x} + \frac{\partial u_2}{\partial y}$$

$$\textit{Gradient}: \qquad \nabla u = \begin{pmatrix} \frac{\partial u_1}{\partial x} \\ \frac{\partial u_2}{\partial y} \end{pmatrix}$$

Using arrays of fields, these operators can be written as Modelica functions, as shown in Figure 2.

The controller is a simple proportional controller checking the temperature at the sensor coordinate, and setting the heater source term proportional to the difference between the actual and the goal temperature. The equation for the controller is:

$$Q_h = c + k * (T_{goal} - T_{sensor}) \qquad (5)$$

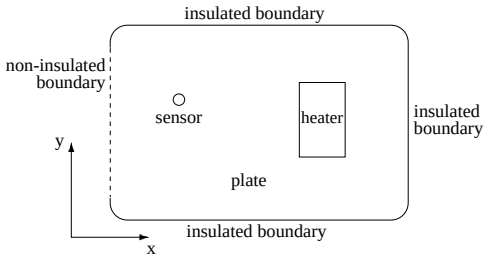


Figure 1: Overview and domain of the example in Figure 3.

Implementation

The field type and field constructors have been implemented in the OpenModelica [1, 13] environment. The type system is extended with the field type including domain information for the field. Some of the operators are also overloaded according to the specification in previous sections. Specifically, the operators **der**() and **pder**() are handled in order to keep the type of the field variable that they are applied to. Also, the domain information, which is kept together with the field type, is preserved in **der**() and **pder**() expressions.

```
function gradient
  input field Real u;
  output field Real v[2];
algorithm
  v[1] := pder(u,x);
  v[2] := pder(u,y);
end gradient;

function divergence
  input field Real u[2];
  output field Real v;
algorithm
  v := pder(u[1],x) + pder(u[2],y);
end divergence;
```

Figure 2: Divergence and gradient operators in two dimensions, defined in terms of the **pder**() operator, using arrays of fields.

Field constructors are implemented by implicitly generating functions with the iterators as the input arguments and the field constructor expression as the function body calculating the return value. Thus, when discretizing, the field calculator function can be called to get the field value at desired points over a given mesh.

Solution

In order to simulate models containing PDEs, the domains are discretized and fields are replaced with arrays of values over the discretized domains. Domain discretization is done by external tools, e.g. a mesh generator when using the finite element method. The PDEs are replaced by a space-discretized set of equations, containing only scalar, time-dependent variables, e.g. the field variables are replaced with arrays of scalar variables. The result is standard Modelica which can be simulated in existing Modelica tools, using methods for solving ordinary differential and algebraic equations. The simulation results can be plotted in three-dimensional plots in tools like Mathematica or Matlab.

```
model ControlledHeater
  import Modelica.SIUnits.Temperature;
  import Modelica.Math.sin;
  TestDomain omega;
  parameter Real T0=0, T0_h=0;
  parameter Real rho=1, rho_h=1, C=1, C_h=1, k=1, k_h=1;
  parameter Real k_n=1, l_n=1;
  Real Q_h;
  Real T_out;
  Real T_sensor;
  parameter Real heateroffset=0, heaterk=1, T_goal=23;

  field Temperature T(domain=omega);

protected
  parameter Real h_n = k_n/l_n;

initial equation
  T = field(T0*sin(x+y) for (x,y) in omega.plate);
  T = T0_h in omega.heater;

equation
  rho*C*der(T) - divergence(k*gradient(T)) = 0 in omega.plate;
  rho_h*C_h*der(T) - divergence(k_h*gradient(T)) = Q_h in omega.heater;

  // Boundary conditions
  k*nder(T) = 0 on omega.ins;
  k*nder(T) + h_n*T = h_n*T_out on omega.nonins;

  // Controller
  T_out = 18 + 8*sin(time);
  T_sensor = T(1.4,1.7);
  // coordinates in the domain, mapping field -> Real
  Q_h = heateroffset + heaterk*(T_goal - T_sensor);

end ControlledHeater;
```

Figure 3: A model of heat transfer in a domain with a heater and a sensor.

Related Work

There are different categories of packages for solving PDEs. Some of them are code libraries, where the PDE is not separately specified but a numerical solver is written using a programming language and components from these libraries in order to solve the specific PDE problem. PETSc [2], Diffpack [3] and Overture [14] are some packages in this category.

There are also problem solving environments, that contain integrated tools for the different steps of the modeling and simulation process, such as graphical tools for defining the domain, tools for specifying or selecting a numerical solver among several solvers, and tools for visualization of the simulation results. PELLPACK [8] is such a problem solving environment that contains several PDE solvers and has a high level language for the PDE problem definition. FEMLAB [5] is another simulation tool, with graphical user interface where the user can choose a model among many predefined PDE models, modify its parameters, graphically define the problem domain and assign boundary conditions, simulate the model and visualize the results.

An environment that is more language oriented, analogous to Modelica, is gPROMS [12]. This environment has a high level language for specifying PDE models on rectangular domains, where complex partial differential and algebraic equations and mixed systems of integral, partial and ordinary differential and algebraic equations can be solved.

In our approach, we define a high-level language for describing PDE models and general geometric domains. The proposed language constructs aim to be consistent with the standard Modelica language constructs, and allow combining of existing Modelica models and new PDE models.

Conclusion and Future Work

In this paper we have described how field variables can be introduced and used in Modelica. Both the syntax and semantics was handled, in terms of declaration of variables and expressions for defining constant fields, and overloading of operators for convenient use of field variables in expressions. We have also described how field variables can be implemented in a Modelica translator, by generating functions that can be used to interpolate fields during the discretization step.

The use of field variables in Modelica models introduces an abstraction level for distributed models that is consistent with the existing models in current Modelica. Continuous domains and fields on continuous domains keeps the model clean from discretization details. If discretization information is necessary to include in the model, this can easily be introduced using annotations, which is a mechanism to add generic, tool or simulation related information to Modelica models in a standardized way.

The field constructor can be used to construct fields and directly use them in expressions. Implicitly generated field constructor functions allows changing the discretization detail without changing the original Modelica model.

Future work involves extracting the partial differential equations and replacing them with discretized ordinary differential equations, using the method of lines approach. Different discretization methods can be used, such as the finite difference, the finite element or the finite volume methods. Field variables are then replaced by array variables. The field access operation for unknown field variables can be implemented by generating a function which interpolates the solution at the given coordinate. Also, the field extensions can be integrated into the simulation backend and the code generation. Using parts of the Rheolef finite element environment [17] is preferred, since this will simplify implementation of discretized fields over given meshes.

References

[1] Peter Aronsson, Peter Fritzson, Levon Saldamli, Peter Bunus, and Kaj Nyström. Meta Programming and Function Overloading in OpenModelica. In *Proc. of the 3rd International Modelica Conference*, pages 431–440, Linköping, Sweden, Nov. 3–4 2003.

[2] Satish Balay, Kris Buschelman, William D. Gropp, Dinesh Kaushik, Lois Curfman McInnes, and Barry F. Smith. PETSc home page. `http://www.mcs.anl.gov/petsc/`, 2001.

[3] Diffpack home page. `http://www.diffpack.com/`.

[4] Dymola. `http://www.dymola.com/`.

[5] FEMLAB home page. http://www.femlab.com/.

[6] P. Fritzson, J. Gunnarsson, and M. Jirstrand. MathModelica - An Extensible Modeling and Simulation Environment with Integrated Graphics and Literate Programming. In *Proc. of the 2nd International Modelica Conference*, Munich, March 2002. http://www.modelica.org/.

[7] Peter Fritzson. *Principles of Object-Oriented Modeling and Simulation with Modelica 2.1*. Wiley-IEEE Press, 2003. http://www.mathcore.com/drmodelica/.

[8] E. N. Houstis, J. R. Rice, S. Weerawarana, A. C. Catlin, P. Papachiou, K.-Y. Wang, and M. Gaitatzes. PELLPACK: A Problem-Solving Environment for PDE-Based Applications on Multicomputer Platforms. *ACM Transactions on Mathematical Software*, 24(1):30–73, March 1998.

[9] MathModelica. http://www.mathcore.com/.

[10] Modelica Association. *Modelica Standard Library*. http://www.modelica.org/libraries.shtml.

[11] Modelica Association. *Modelica – A Unified Object-Oriented Language for Physical Systems Modeling - Language Specification Version 2.1*, Jan 2004. http://www.modelica.org/.

[12] M. Oh. *Modelling and Simulation of Combined Lumped and Distributed Processes*. PhD thesis, University of London, 1995.

[13] OpenModelica. http://www.ida.liu.se/~pelab/modelica/OpenModelica.html.

[14] Overture home page. http://www.llnl.gov/CASC/Overture/.

[15] Python. http://www.python.org/.

[16] Levon Saldamli. *PDEModelica - Towards a High-Level Language for Modeling with Partial Differential Equations*. Linköping Studies in Science and Technology, Licentiate Thesis No. 990, December 2002.

[17] Pierre Saramito, Nicolas Roquet, and Jocelyn Etienne. Rheolef home page. http://www-lmc.imag.fr/lmc-edp/Pierre.Saramito/rheolef/, 2002.

Proceedings of SIMS 2004
Copenhagen, Denmark
September 23–24, 2004

MODELLING, SIMULATING AND OPTIMIZING BOILERS

Kim Sørensen
Aalborg University, Institute of Energy Technology, Pontoppidanstræde 101, DK - 9220 Aalborg
Tel: +45 96 35 92 48, Fax: +45 98 15 14 11, kso@iet.auc.dk, http://www.iet.auc.dk/
Aalborg Industries A/S, Gasværksvej 24, P.O. Box 661, DK-9100 Aalborg, Denmark
Tel: +45 99 30 45 43, Fax: +45 99 30 44 53, kso@aalborg-industries.dk, http//www.aalborg-industries.dk/

Thomas Condra
Aalborg University, Institute of Energy Technology, Pontoppidanstræde 101, DK - 9220 Aalborg
Tel: +45 96 35 92 65, Fax: +45 98 15 14 11, tc@iet.auc.dk, http://www.iet.auc.dk/

Niels Houbak
Technical University of Denmark, MEK - Energy Engineering Section,
Nils Koppels Alle' 402, DK - 2800 Kgs. Lyngby
Tel: +45 45 25 41 54, Fax: +45 45 93 06 63, niels.houbak@mek.dtu.dk, http://www.et.dtu.dk/

ABSTRACT

In the present work a framework for optimizing the design of boilers for dynamic operation has been developed. A cost function to be minimized during the optimization has been formulated and for the present design variables related to the Boiler Volume and the Boiler load Gradient (i.e. firing rate on the boiler) have been defined. Furthermore a number of constraints related to: minimum and maximum boiler load gradient, minimum boiler size, *Shrinking and Swelling* and *Steam Space Load* have been defined.

For defining the constraints related to the required boiler volume a dynamic model for simulating the boiler performance has been developed. Outputs from the simulations are shrinking and swelling of water level in the drum during for example a start-up of the boiler, these figures combined with the requirements with respect to allowable water level fluctuations in the drum defines the requirements with respect to drum size.

The model has been formulated with a specified *building-up* of the pressure during the start-up of the plant, i.e. the steam production during start-up of the boiler is output from the model. The steam outputs together with requirements with respect to steam space load have been utilized to define requirements with respect to steam drum size.

The cost function has been defined as a function taking the weight of the boiler, its dynamic capability and finally the consumptions during dynamic operation of the boiler into consideration (the latter presumed constant). It has been shown that the weight of the boiler is (with an acceptable accuracy) proportional with the volume of the boiler. For the dynamic operation capability a cost function penalizing limited dynamic operation capability and vise-versa has been defined. The main idea is that it by mean of the parameters in this function is possible to fit its shape to the actual application.

In the paper an optimization example is shown and the results discussed. By means of the developed model it is shown how the optimum *changes* from a boiler favoring a good dynamic capability (i.e. a boiler with a relatively large volume) to a boiler not penalizing a limited dynamic capability (i.e. a boiler with a relatively small volume) if the shape of the cost function is changed slightly.

Keywords: Optimization of dynamic boiler performance. Dynamic boiler modelling and simulation, drums, DAE and MATLAB.

NOMENCLATURE

Symbol	Description	Unit
A	Area	[m^2]
F	Objective function	[−]
M	Mass	[kg]
NW	Normal Water level	[−]
P	Power	[J]
p	Pressure	[bar]
T	Temperature	[deg C/or K]
V	Volume	[m^3]
$\mathbf{X}$	Design Variable	[−]

Subscript	Description
boi	boiler
cons	consumption
dr	drum
dyn op	dynamic operation
s	steam
sat	saturation

INTRODUCTION

Since the first boilers for *transforming* fossil energy (solid fuels, oil or gas) to the energy carrying medium (water, steam, thermal fluid etc.) were developed at the start of the industrial era, the advanced heat exchangers, as boilers actually are, have been the subject of continuous development aiming at: higher efficiency, lower emission levels, higher availability[1], better operation performance etc. During this period the focus of the development has been strongly influenced by the surroundings (higher energy costs causing request for higher efficiency, environmental attention causing request for lower emissions, higher salary levels causing request for simpler operation etc.).

Characteristic for all the dimensions is that the development has been kind of *asymptotic*, i.e. boiler efficiency $\rightarrow$ 100 % (state of the art: 94 - 95 %), emissions $\rightarrow$ 0, availability $\rightarrow$ 100 % etc.

The continuous development towards more and more efficient plants with more and more advanced steam data has resulted in boilers that today are among the largest manmade steel constructions on earth.

Traditionally boilers have been developed and optimized for static operation and limited attention has been paid to the boilers dynamic performance - typically the boilers dynamic performance has been analyzed to avoid damaging the boiler (e.g. over stressing the materials due to high temperature gradients). Today the requirements with respect to for example emissions, efficiency, fuel flexibility and dynamic performance are changing dramatically.

[1] A plants availability is defined as the *time* the plant is available for operation divided by the total *time*.

CHALLENGES OPTIMIZING BOILERS FOR DYNAMIC OPERATION

Increased requirements, with respect to dynamic performance, have a number of built-in opposing aims (contradictions):

- Drum size - stress level. Both natural circulating and once-through boilers need a reservoir for absorbing shrinking and swelling during the dynamic operation of the boiler (for example start-up or a sudden load change). On the one hand, to be able to absorb the fluctuations within the boiler a large reservoir is required/desirable. On the other hand, the material thickness in a pressurized vessel (the reservoir) is approximately proportional to the diameter, i.e. the higher pressure, the larger material thickness. However the allowable temperature gradients for the pressurized vessel decrease as the wall-thickness increases (the stresses introduced in the thick-walled boiler constructions related to temperature gradients are approximately proportional to the square of the material thickness) - see [10].

- Drum size - steam quality. Depending on the exploitation of the boilers steam production different requirements with respect to steam quality (i.e. *dryness*) are defined. In general the better steam quality required, the smaller *carry-over* is accepted. The requirements with respect to steam quality are closely related to the size of the *steam space* (i.e. boiler drum). Depending on the boiler plants *operation philosophy* the requirements with respect to steam quality will define the size of the drum. A quick start-up (or load change) on the boiler will therefore require a relatively large drum, which on the other hand limits the allowable gradients on the plant.

- Control system. Depending on the complexity

of the boiler control system the water level fluctuations can be controlled (i.e. limited) meaning that the required dynamic performance can be obtained with a smaller (i.e. cheaper) boiler. On the other hand a more complex control system is a larger investment and typically the operation costs are correspondingly higher.

- Drum size - start-up gradient. For most boiler plants the dynamic performance is closely related to the allowable gradients in the thick-walled boiler constructions. For boilers producing saturated steam the temperature gradients are closely related to the pressure gradient, $dT_{sat}/dp = f(p_{sat})$, meaning that the pressure gradients define requirements with respect to the wall thickness and hereby the volume of the boiler - see [10].

- Dynamic vs. static load conditions. Depending on the application of the boiler plant the importance of the boilers dynamic capability can vary significantly. For some applications (for example *stand-by* boilers foreseen to start producing steam very fast in special situations) a unique dynamic performance is required and it therefore possesses a high value. For other applications the dynamic performance is of minor importance (for example *base load plants* foreseen to be operating at stationary load most of the time).

- Boiler construction and choice of materials. Depending on the requirements to the boiler plant with respect to weight, height, foot print etc. opposing aims within the *detailed engineering* to design the cheapest boiler will be present. For example boiler constructions based on *few long tubes* will be cheaper than boiler constructions based on *many short tubes* (fewer weldings etc.). Furthermore the optimization could be extended to include exploitation of more *advanced materials*, i.e. alloyed materials with better material properties, e.g. higher allowable stresses. If this dimension is included in the optimization the different manufacturing technologies applied for the different materials should also be included[2].

[2] Actually this dimension would require deeper in-sight into the different competence levels at different manufacturing locations, e.g. the Far East versus Western Europe.

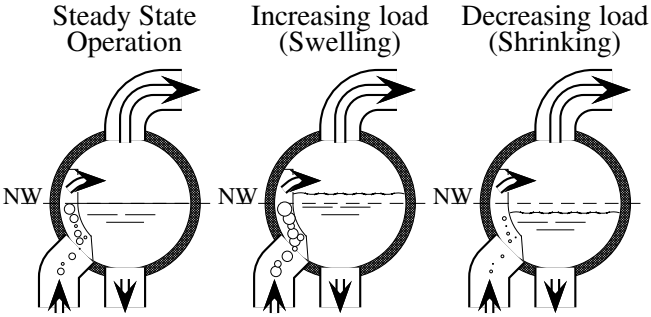


Figure 1: Shrink and swelling in the steam drum during increased and decreased load.

The optimization of the complete boiler concept could also include the operation conditions, e.g. number of operation hours at full load and at part load.

And finally:

- Quantification of the boilers dynamic performance. The requirements with respect to dynamic operation of the boiler plant (i.e. increasing the load on the plant) have to be quantified to determine to what extent increased capability with respect to dynamic operation can justify investments in boilers with better dynamic operation capability, i.e. shorter start-up time, larger load-gradients etc. - see [9].

Shrinking and Swelling

As described the water level in the drum will fluctuate due to changes in the pressure in the boiler and the heat input to the evaporator (see Figure 1). For *lower pressure levels* the volume of the steam fraction in the evaporator will be relatively large causing larger fluctuations as a result of the changes in heat input or pressure level. For *higher pressure levels* the variations will be relatively smaller.
As the *feed water controllers* main purpose is to control the water level in the boiler drum, the design of the feed water controller may (strongly) affect the amount of shrinking and swelling.
Shrinking and swelling is closely related to the firing rate on the boiler (see Figure 1), i.e. increased firing rate gradient will *compress* the time interval

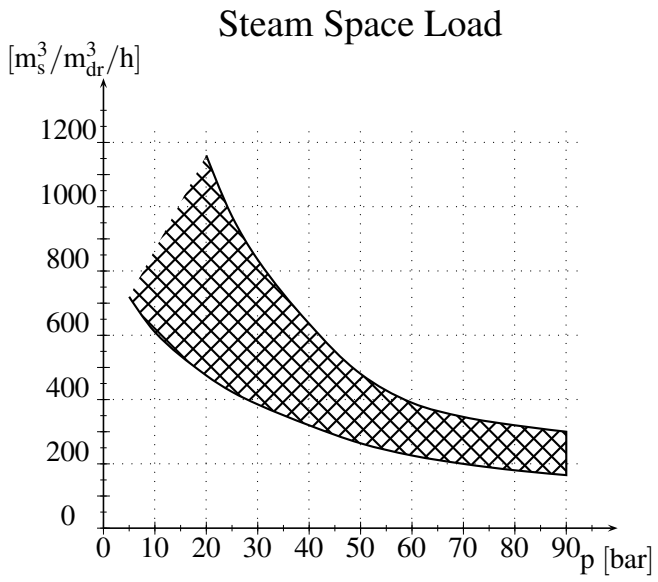


Figure 2: Steam space load - see [1].

where the shrinking and swelling affects the water level causing larger fluctuations in water level.

The fluctuations in water level are closely linked to the volume of the drum and hereby its wall thickness - see [8]. This means that there is a close link between the shrinking and swelling and the allowable temperature gradients.

Steam Space Load

A boiler is normally designed with a certain water/steam volume to be able to absorb the fluctuations in the water level during dynamic operation of the plant. Operating the boiler the *steam space load* has to be taken into consideration. As a rough rule of thumb for a certain quality (i.e. dryness) of the steam leaving the boiler, requirements can be put on the boiler drums steam volume in relation to the total steam production, i.e. $\left[\mathrm{m_s^3}/(\mathrm{h}\cdot\mathrm{m_{dr}^3}) = 1/\mathrm{h}\right]$. This figure is normally given as a band between 2 curves (see Figure 2) specifying the required steam space[3]. The shape of the curve(s), i.e. higher pressure correspond to lower specific steam space load, is related to the fact that the higher pressure, the smaller difference between the specific weight of water and steam. At this stage no relation between the steam production and the water space/volume can be given[4].

[3]The band illustrates the uncertainty on the required steam space load, where for example salt content and efficiency of water/steam separation in the drum has to be taken into consideration.

OPTIMIZATION PROCEDURE

An optimization problem is characterized by having an objective function to be minimized[5]:

$$\textit{minimize} \qquad F(\mathbf{X}) \qquad \text{objective function}$$

which is subject to the constrains:

$$h_i(\mathbf{X}) = 0, \qquad i = 1,2,\ldots,I \qquad \text{equality constraints}$$

$$g_j(\mathbf{X}) \geq 0, \qquad j = 1,2,\ldots,J \qquad \text{inequality constraints}$$

where

$$\mathbf{X} = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \end{bmatrix}$$

is the design variable.

This procedure will be applied in the actual optimization process.

As described optimizing the design of boilers (in the present study) from a dynamic operational point of view is a matter of on the one hand minimizing the volume of the boiler, (X_2), still being able to handle the shrinking and swelling of the water/steam mixture and meet the requirements with respect to steam space load and at the same time maximizing the load gradient on the boiler, (X_1).

This optimization will be carried out specifying an *objective function* as a function of the 2 design variables, $\mathbf{X}^T = (X_1, X_2)$, and by means of a dynamic model for the boiler defining the constraints related to shrinking/swelling and the steam space load. After having delimited the *feasible set* by means of the constraints, the optimum can be found.

The objective function will include:

[4]For the water volume no requirements are specified (in principle *once-through* boilers do not have a water reservoir), but in general the water accumulated in the boiler drum is the greatest thermal buffer in the system, i.e. the *stabilizer* of the system pressure. Requirements with respect to water volume are related to safety aspects, e.g. the boiler must not *dry out* due to lack of water.

[5]Without loss of generality we always operate with an objective function to be minimized. If an objective function $F(\mathbf{X})$ has to be maximized, $-F(\mathbf{X})$ has to be minimized.

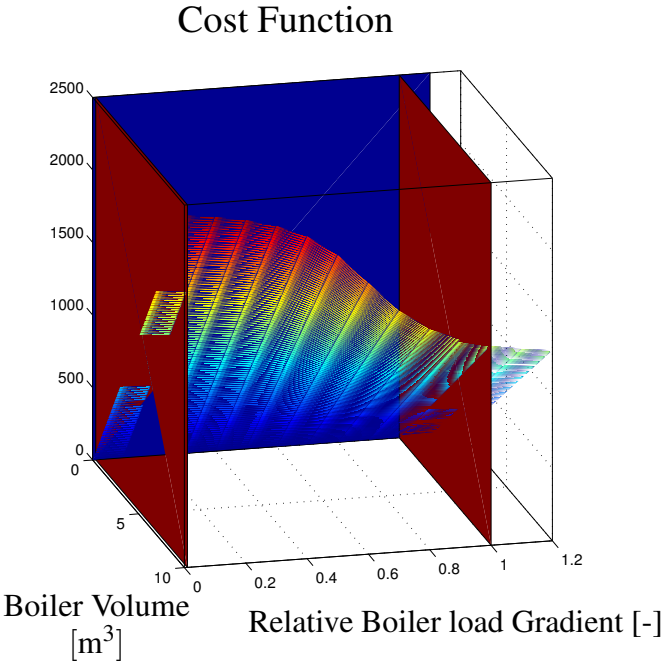


Figure 3: Cost Function (see [9]) with constraints (Equation 2 and 3).

- the investment costs
- the operational costs, where the value/quantification of a given dynamic performance is included
- the consumptions related to the dynamic operation of the boiler plant.

The following objective function has been applied for the analysis:

$$F(\mathbf{X}) = F_{mass} + F_{dyn\ op} + F_{cons}. \tag{1}$$

Where:

- $F_{mass} = M_{boi} \cdot$ *specific cost of materials*
- $F_{dyn\ op}$ = quantified value of dynamic performance
- F_{cons} = quantified value of the consumptions during dynamic operation of the plant.

As the optimization task is to minimize the price[6] of the boiler, the following design variables have been chosen for the optimization:

$$\mathbf{X} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \left(\frac{\frac{dP_{boi}}{dt}}{\frac{dP_{boi,max}}{dt}}\right) \\ V_{boi} \end{bmatrix} = \begin{bmatrix} \left(\frac{dP_{boi}}{dt}\right)_{rel} \\ V_{boi} \end{bmatrix} =$$

$$\begin{bmatrix} Relative\ Boiler\ Load\ gradient \\ Volume\ of\ boiler \end{bmatrix}$$

As described a feasible set has to be defined/delimited by means of a number of constraints, which for the present optimization task are inequality constraints. For the design variable V_{boi}, the following (simple) constraint can be defined:

$$V_{boi} \geq 0 \tag{2}$$

and for the design variable $(dP_{boi}/dt)_{rel}$, the following (simple) constraints can be defined:

$$0 \leq \left(\frac{dP_{boi}}{dt}\right)_{rel} \leq 1 \tag{3}$$

Furthermore the volume of the steam drum has to be checked against the steam space load (see Figure 2) and the requirements with respect to shrinking and swelling due to dynamic operation.
For defining the constraints related to shrinking and swelling and steam space load a dynamic model of the boiler has been developed. The dynamic model is described in detail in [8], more detailed dynamic models are described in [12], [6], [5] and [4].

OPTIMIZATION OF WATER TUBE BOILER DESIGN FOR DYNAMIC OPERATION

For the different *Boiler load Gradients* the required steam space in the boiler drum has been calculated on the basis of the steam production. The allowable *steam space load* during start-up has been defined as the double of the allowable steam steam space load during steady state operation, i.e. 1.800 $[m_s/h/m_{dr}]$[7]. For calculating the required drum volume to meet the requirements with respect to steam space load it is presumed that the steam volume is 50 % of the drum volume.

[6] Price is seen as a broader term including quantification of other important parameters, see [9].

[7] During the start-up of the plant, where the pressure is building up according to the selected operation conditions, the operation data are almost *out of range* according to Figure 2. Extrapolating the data to the lower pressures indicates that even higher steam space load could be allowed especially for a shorter period.

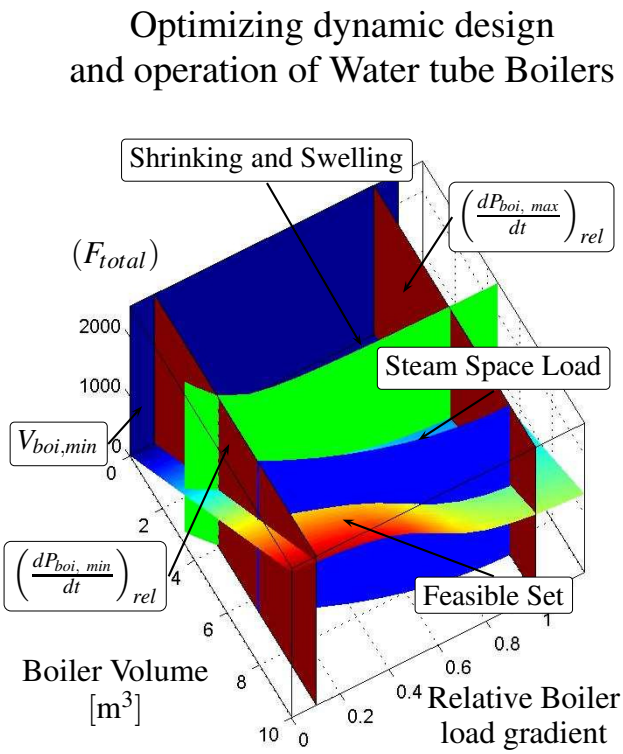


Figure 4: Cost Function with *Feasible Set* defined by means of constraints (Equation 2 and 3) and constraints related to *Shrinking and Swelling* and *Steam Space Load*.

A plot of the *Cost Function* as a function of the 2 design variables, $\mathbf{X}^T = \left[\ V_{boi}, (dP_{boi}/dt)_{rel}\ \right]$, is shown in Figure 4. In the figure the constraints related to the minimum drum volume (Equation 2), minimum and maximum boiler load gradients (Equation 3) define the feasible set (see also Figure 3).

Furthermore the constraint related to shrinking and swelling and the constraint related to allowable steam space load are shown in Figure 4.

For the simulations it has been presumed that a specific boiler load gradient lower than 10 % would not be relevant, (i.e. $(dP_{boi}/dt)_{rel} \geq 10\%$).

The results from the optimization can also be seen in the contour plot in Figure 5.

As can be seen from Figure 4 and 5 the global minimum of the cost function can be found at the intersection between the maximum allowable boiler load gradient and the maximum allowable steam space load[8]. Furthermore it can be seen from Figure 4 and

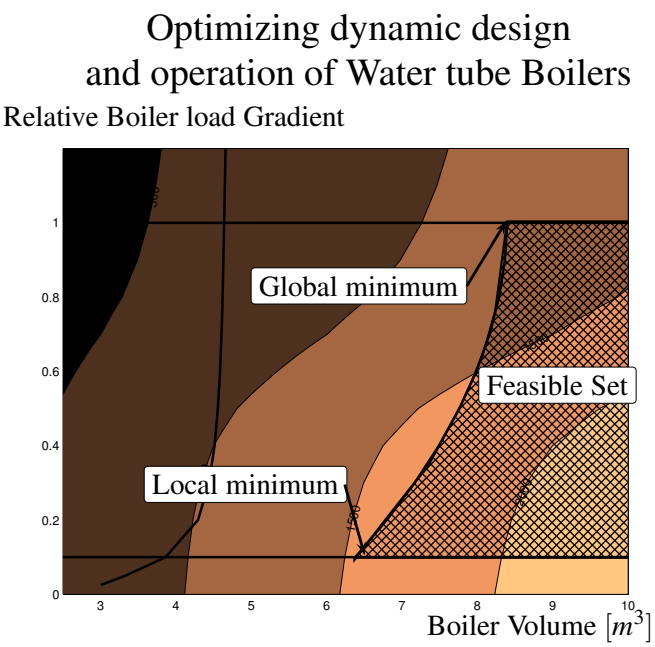


Figure 5: Contour plot of Cost Function with *Feasible Set* defined by means of constraints (Equation 2 and 3) and constraints related to *shrinking and swelling* and *steam space load*.

5 that a local minimum of the objective function is located at the intersection between the minimum allowable boiler load gradient and the maximum allowable steam space load.

With the *chosen* parameters in the quantification of the boilers capability with respect to dynamic operation (see [9]) the surface of F_{total} in Figure 4 and 5 is *rather flat* in the *Relative Boiler load Gradient* direction, i.e. the optimum design could by minor changes of the equation parameters *move* to the *local minimum* (see Figure 5). This change of optimum in the direction of a cheaper boiler with lower dynamic capability is the obvious development if the quantification of dynamic performance is *low*. For many practical applications the requirements with respect to dynamic performance will be rather *rigid*, e.g. *the plant must be able to change load by 20 % per minute*. In these situations the constraint with respect to minimum allowable boiler load gradient, $(dP_{boi,min}/dt)_{rel}$, would be located at the specified value. Alternatively the parameters in the objective function should be chosen to *simulate* a discontinuity at the minimum required boiler load gradient - see [10] for a more detailed discussion.

Depending on the shape of the quantification of

[8] Attention should be drawn to the fact that the quantification of the boilers capability with respect to dynamic operation is based on the shape/course of the Cost Function as defined in [9].

the boilers dynamic capability (see [10]), the global minimum can change/move. If for example a rather low boiler load gradient is acceptable for the actual application, this would correspond to a specific choice of parameters in the objective function (see [9] and [10]), lowering the surface (see Figure 4 and 5) in the region with the local minimum, and at a certain point turn this into the *global minimum*[9].
As can be seen from Figure 4 and 5 the requirements/constraints with respect to steam space load are more *restrictive* than the requirements/constraints with respect to shrinking and swelling. *Loosening* the requirements with respect to steam space load[10] would *move* this constraint to the left in Figure 5. At a certain stage the 2 constraints would *cross* each other. But still (with the selected operation conditions) the global minimum of the cost function would be found at the intersection between the maximum allowable boiler load gradient and the requirements with respect to maximum allowable steam space load or the requirements/constraints with respect to shrinking and swelling.

CONCLUSION

In this paper a framework for optimizing boiler designs for dynamic operation has been given. The optimization has been developed *open*, i.e. more optimization challenges and corresponding design variables can be included. The *feasible set* for the optimization has been delimited by simple constraints related to the boiler volume and to the minimum and maximum boiler load gradients. Furthermore constraints have been defined on the basis of dynamic simulation of the boiler performance for *steam space load* and *shrinking and swelling*.
For the chosen parameters in the optimization's *objective function* quantifying the boilers capability with respect to dynamic performance the (global) optimum design is a design based on a boiler designed for the maximum allowable boiler load gradient. The optimization shows that a *local* optimum is present for a design with the minimum allowable boiler load gradient, i.e. a boiler with a small volume. The optimization further shows that minor changes in the quantification of the boilers dynamic capability will *change* the local optimum to the global optimum and vise-versa, i.e. favoring the small boiler design with the minimum dynamic operation capability.

[9] This corresponds to a situation where the plant for example is foreseen to run at base load all the time.

[10] For example increasing the requirements with respect to allowable salt content in the boiler water - see Figure 2.

PERSPECTIVES

For future studies within *Dynamic Boiler Performance - modelling, simulating and optimizing boilers for dynamic operation* the following topics are recommended:

- selection of materials for boilers to optimize dynamic behavior further - to include manufacturing optimization
- validation of the *Cost Function* developed in the present study - to include optimizations with large discontinuities in for example material or manufacturing prices
- analysis including different operational patterns
- sensitivity analysis of the model developed in the present study - to include the *Cost Function* and the dynamic models
- *maturing* of the developed model, i.e. preparation for *practical use*
- include *long term* respectively *short term* optimization in the model
- *stochastic models*, e.g. life time analysis.

REFERENCES

[1] *Basic Design Manual - Aalborg Industries A/S*, 2004 - proprietary.

[2] *Dubbel - Taschenbuch Für den Maschinenbau*, 13. Auflage, neudruck, Springer-Verlag Berlin Heidelberg New York, 1974.

[3] *Homepage of "Aalborg Industries A/S"* `http://www.aalborg-industries.com/`

[4] Doležal, R., Kammer, G. v.d. & Königsdorf, E.: *Simulation des Anfahrens eines Dampferzeugers mit einem universalen, für große Zustandsübergänge geeigneten Modell*, VGB Kraftwerkstechnik 55, Heft 8, November 1975.

[5] Mayer, Uwe: *Simulation und Analyse des Abfahrverhaltens eines 765-MW-Kombikraftwerkes mit einem semianalytischen entkoppelten Rechenmodell*, Ph.D. Thesis, Fortschritt-Berichte VDI, Reihe 6, Nr. 249, 1990.

[6] Rolf, Albrecht: *Simulation des Nichtlinearen, dynamischen Verhaltens von Värmetauschern sowie ihrer komplexen Schaltungen im Kraftwerksbau mit einem semianalytischen Berechnungsverfahren.*, Dissertation, Institut für Verfahrenstechnik und Dampfkesselwesen der Universität Stuttgart, 1984.

[7] Strauß, K.: *Kraftwerks-technik zur Nutzung fossiler, regenerativer und nuklearer Energiquellen*, Springer, 1998.

[8] Sørensen, Kim; Condra, Thomas & Houbak, Niels: *Modelling of boiler heating surfaces and evaporator circuits* Presented at SIMS-Scandinavian Simulation Society, 43rd SIMS Conference (SIMS 2002), University of Oulu, Finland, September 26 - 27, 2002.

[9] Sørensen, Kim; Houbak, Niels & Condra, Thomas: *Modelling, Simulating and Optimizing Boilers* Presented at, The 16th International Conference on Efficiency, Costs, Optimization, Simulation and Environmental Impact of Energy Systems (ECOS 2003), Technical University of Denmark, June 30 - July 2, 2003.

[10] Sørensen, Kim; Houbak, Niels & Condra, Thomas: *Modelling, simulating and optimizing boiler heating surfaces and evaporator circuits* Presented at SIMS-Scandinavian Simulation Society, 44th SIMS Conference (SIMS 2003), Västerås, Sweden, September 18 - 19, 2003.

[11] *TRD - Technische Regeln für Dampfkessel*, Verband der Technischen Überwachungs-Vereine e.V., Carl Heymanns Verlag KG, 1998.

[12] Walter, H.: *Modellbildung und numerishe Simulation von Naturumlaufdampferzeugern*, Ph.D. Thesis, TU-Wien, Fortschritt-Berichte VDI Nr. 457, Wien, Mai 2001, ISBN 3-18-345706-7.

Author Index

Aldini, Lorenzo, 3
Andersen, Stig Kildegård, 119
Andrei Vijeu, Razvan, 129
Aronsson, Peter, 389
Atanassov, Krassimir T., 351
Aurdal, Lars, 373

Bellettre, Jerome, 129, 137, 233, 397
Bellini, Anna, 7
Berta, Gian Luigi, 341
Blando, Gandolfo, 163
Boldt, Niels, 291, 315
Bunus, Peter, 381

Cakir, Kemal, 357
Carlsen, Henrik, 119
Casoli, Paolo, 341
Christensen, Søren Tørholm, 299
Christova, Nikolinka G., 351
Condra, Thomas, 429

D'Ayala, Dina, 281
Dahlquist, Erik, 53, 89, 103
Damsgaard, Michael, 299
de Zee, Mark, 299
Dhak, Janice, 89
Djupesland, Per G., 373
Dohlmann, Henrik, 307

Erleben, Kenny, 307, 323

Faber, Richard, 95
Franzoni, Germano, 341
Fritzson, Peter, 381, 389, 405, 421
Furenes, Beathe, 17

Garnier, Cedric, 233
Gerun, Luc, 137
Guglielmino, Emanuele, 281
Guney, Baha, 357
Guyonvarch, Gwénaël, 233

Gündogar, Emin, 357

Hald, John, 69
Hamidipour, Mohsen, 27
Hattel, Jesper, 7, 35, 69
Hladnik, Ales, 45

Iliev, Hristo, 273

Jansson, Johan, 103
Johansson, Olof, 405
Jonsdottir, Fjola, 155
Jonsson, Magnus Thor, 155
Juslin, Kaj, 59
Juuso, Esko, 145

Kangas, Petteri, 59
Kappen, Johannes, 53
Karlsdottir, Anna, 155
Karstensen, Claus M.S., 365
Kim, Sung-Kwon, 215
Kleven, Marit, 373
Korsgaard, Anders, 181
Kovacevic, Ahmed, 241
Kundzina, Antra, 223
Kær, Søren Knudsen, 181

Lappalainen, Jari, 59
Lauenroth, Thomas, 79
Li, Bo, 95
Li, Pu, 95
Lie, Bernt, 17
Lo Mastro, Fabio, 163
Lundvall, Håkan, 381

Maccioni, Raffaele, 3
Manninen, Jussi, 53
Melaaen, Morten, 373
Meyer, Jonas, 315
Mostoufi, Navid, 27

Naghashzadegan, Mohammad, 173, 265
Negro, Carlos, 53
Niels, Houbak, 429
Nielsen, Mads Pagh, 181
Nikian, Mohammad, 173, 265
Nyström, Kaj, 389

Ollivier, Eric, 397

Pedersen, Camilla, 323
Persson, Tommy, 189, 199
Pop, Adrian, 405
Preisig, Heinz A, 413
Proykova, Ana, 273
Pryds, N.H., 35
Pulkkinen, Petteri, 111
Pursimo, Juha, 59

Rasmussen, John, 299
Reimers, Martin, 373
Ritala, Risto, 53, 111
Rørbech, Marinus, 331
Røtnes, Jan Sigurd, 373

Saldamli, Levon, 421
Savolainen, Jouni, 59
Shin, Jee-Young, 207, 215
Sireteanu, Tudor, 281
Slonevsky, Roman, 249
Smith, Ian. K, 241
Son, Young-Seok, 207, 215
Sotudeh-Gharebagh, Rahmat, 27
Sporring, Jon, 323
Stammers, Charles, 281
Stancioiu, Danut, 281
Stark, Helmut, 45
Stolyarchuk, Roksolana, 249
Stosic, Nikola, 241
Stravs-Grilc, Barbara, 45
Surma, Egidijus, 299
Sørensen, Kim, 365, 429

Taymaz, Imdat, 357
Tazerout, Mohand, 129, 137, 233, 397
Thomsen, Per Grove, 119
Thorborg, Jesper, 7, 69
Turlajs, Daniels, 223

Vacca, Andrea, 341
Vardar, Nurten, 255
Vostrikovs, Sergejs, 223
Vîjeu, Rãzvan, 137

Weiss, Dietmar, 79
Wollerstrand, Janusz, 189, 199
Wozny, Günter, 95

Ylén, Jean-Peter, 59
You, Byoung-Mu, 207

Keyword Index

Action E36, 53
Air Conditioning System, 265
Aluminium, 7
Artificial Neural Networks, 45
Atomization, 35

Balance Strategy, 323
Biomass, 129
Biomechanics, 323
Boiler Dynamic, 365
Buried Pipes, 189

Capacitor Discharge Stud Welding, 79
Car Heating, 233
Cavitation, 341
Cold Start, 233
Collision Detection, 315
Combustion Modelling, 233
Component-based Modelling, 421
Compressible Flow, 119
Computational Fluid Dynamics, 241, 373
Computer Software, 265
Computer-aided, 413
Constant-composition Solid-solid Phase Transformations, 273
Contact Analysis, 307
Contact Graphs, 307
Contact Groups, 307
Control Mechanism, 323
Control Volume, 119
Conventional Baffle Plate, 215
Cooling Load Calculation, 173
Cost, 53
Counter-flow Cooling Tower, 207
Creep, 7, 69

DAE, 413
Data Reconciliation, 95
Declarative Modelling, 421
Deformable-bodies, 291
Deposition, 35
Design Condition, 207
Design System, 155
Discharge Profile Optimization, 79
Distributed Parameter Systems, 145
Domestic Hot Water Circulation, 199
Dynamic Model, 199
Dynamic Modelling, 145
Dynamic Simulation, 357

Earthquake, 281
Elasticity and Inelasticity, 273
Energy Balance, 79

FEM, 69
Finite Element Method, 189
Flow Calculation, 199
Fluent, 373
Fluid Dynamics, 331
Fluid Model, 341
Fuel Cells, 181
Fuel Oil Tank, 255
Fuzzy Set Systems, 145

Gas to Melt Ratio, 35
Generalized Nets, 351
Geothermal Piping Systems, 155
GPU, 331
Graphics Hardware, 315
Grid Computing, 389
Gridmodelica, 389

Heat Exchanger Network(hen) Synthesis, 181
Heat Exchanger Performance, 215
Heat Flows, 189
Heterogenous Reaction, 27
High Level Control, 307
High Pressure Steam Valve, 69
High Temperature, 7
High Viscosity, 255

Image-space, 315
Integrated Modelling, 69
Interaction, 291
Inverse Dynamics, 299

Linguistic Equations, 145
Liquid-bodies, 291

Magnetorheological Damper, 281
Mathematical Modelling, 241
MINLP, 181
Model Based Parameter Estimation, 95
Model Checking, 381
Modelica, 381, 389
Modelling, 35, 45, 53, 129, 163, 223, 351, 413
Molecular Dynamics Simulations, 273
Multibody Dynamics, 299, 307
Muscles, 299
Musculoskeletal Models, 299

Nanoscale Pattern Formation, 273
Navier-stokes, 331
Noise Analysis, 265
Nonlinear Modeling, 365
Nonlinear Models, 145
Nose, 373
Numerical, 255

Object-oriented Modelling, 421
Off-design Performance, 207
One Dimensional, 119
Online Optimization, 95
Open Modelica, 389
Operation Characteristics, 207
Optimization, 111

Paper Properties, 45
Papermaking, 45
Parameter Estimation and Simulink, 365
Parameter Sweep, 389
Partial Differential Equations, 421
Particle Based, 291
PEM Fuel Cell, 163
Plasticity, 69
Process Control, 45
Process Systems Engineering, 413
Process Window, 35
Prognosis, 223
Pulp and Paper, 53
Pump, 341
Pyrolysis, 129

Railroad Vehicle Simulation, 357
Reactor Modelling, 27
Real-time, 331
Rigid-bodies, 291

Screw Compressor, 241
Screw Expander, 241
Self Intersections, 315
Sequential Modular Approach, 27
Shaded Wall, 173
Shell-and-tube Heat Exchanger, 215
Simulation, 53, 111, 119, 163, 281, 291, 365
Simulation Software, 299
Simulation Tool, 233
Simuling, 199
Single-wall Carbon Nanotubes, 273
Solar Power Plant, 145
Solar Radiation, 173
Spiral Baffle Plate, 215
Spray Forming, 35
Steam Reforming, 181
Stochastic, 111
Storehouse Processes, 351

Time-dependent Model, 189
Turbulent, 255

Unified Constitutive Model, 7
Urea Reactor, 27

Variable Structure Control, 281
Viscoplasticity, 7

Weight Shift Strategy, 323