BOOK OF ABSTRACTS

10th INTERNATIONAL MODELICA CONFERENCE

March 10-12, 2014
Lund, Sweden
www.modelica.org

EDITORS: HUBERTUS TUMMESCHEIT AND KARL-ERIK ÅRZÉN

Modelon

LUND LINNAEUS

MODELICA

The Conference is organized by Modelon in collaboration with the Linnaeus center LCCC at Lund University in cooperation with the Modelica Association.

Book of Abstract of the 10th International Modelica Conference
Lund, Sweden, March 10-12, 2014

corresponding to the

Proceedings of the 10th International Modelica Conference
Lund, Sweden, March 10-12, 2014

Editors:
Dr. Hubertus Tummescheit and Prof. Karl-Erik Årzén

Published by:
Modelica Association and Linköping University Electronic Press

ISBN: 978-91-7519-380-9
Series: Linköping Electronic Conference Proceedings, No 96
ISSN: 1650-3686
eISSN: 1650-3740
DOI: http://dx.doi.org/10.3384/ecp14096

Organized by:

Modelica Association	and	Modelon AB
c/o PELAB, Linköpings Univ.		IDEON Science Park
SE-581 83 Linköping		SE-223 70 Lund
Sweden		Sweden

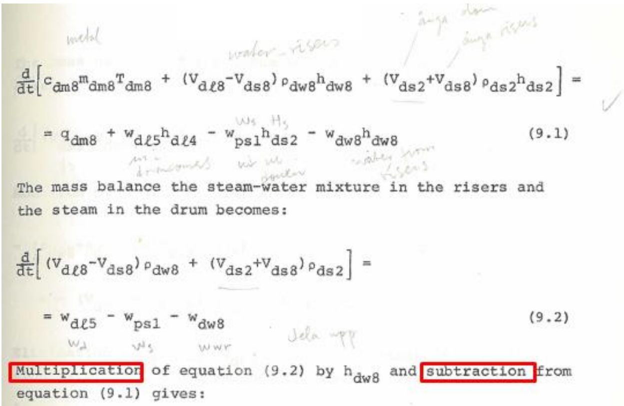
Conference location:
Lund University
Matematikcentrum/Matteannexet
(Center for Mathematical Sciences)
Sölvegatan 20A, SE-223 62 LUND
SWEDEN

Contents

Modelica Evolution – From My Perspective

Hilding Elmqvist
Dassault Systèmes
Ideon Science Park, SE-223 70 Lund, Sweden
Hilding.Elmqvist@3DS.com

This paper intends to tell the story of Modelica (www.Modelica.org) from the author's perspective. It is a fantastic saga that for me started in April 1976. The saga includes studying the needs, the original idea, the development of a solution, waiting for mature hardware technology, a start-up company, a fantastic collaboration, an automotive company caring for its software supplier, how to get momentum by standards collaboration, forming the right team, and the magic phone call from the right company.



$$\frac{d}{dt}\left[c_{dm8}m_{dm8}T_{dm8} + (V_{d\ell 8}-V_{ds8})\rho_{dw8}h_{dw8} + (V_{ds2}+V_{ds8})\rho_{ds2}h_{ds2}\right] =$$

$$= q_{dm8} + w_{d\ell 5}h_{d\ell 4} - w_{ps1}h_{ds2} - w_{dw8}h_{dw8} \qquad (9.1)$$

The mass balance the steam-water mixture in the risers and the steam in the drum becomes:

$$\frac{d}{dt}\left[(V_{d\ell 8}-V_{ds8})\rho_{dw8} + (V_{ds2}+V_{ds8})\rho_{ds2}\right] =$$

$$= w_{d\ell 5} - w_{ps1} - w_{dw8} \qquad (9.2)$$

$\boxed{\text{Multiplication}}$ of equation (9.2) by h_{dw8} and $\boxed{\text{subtraction}}$ from equation (9.1) gives:

Modelica: Systems Engineering, Technology Readiness & Industrial Opportunities

Clas A. Jacobson
Chief Scientist
Corporation Systems & Controls Engineering
United Technologies

The talk focuses on systems engineering in the industrial context. Systems engineering means analytical and in the context of product development. Four elements characterize systems engineering: (1) requirements, (2) architecture, (3) model based development and (4) design flows. Lessons learned from using Modelica in several industrial case studies will be reviewed. Industrial needs will be presented for wider & deeper deployment of Modelica.

The Functional Mockup Interface - seen from an industrial perspective

Christian Bertsch Elmar Ahle Ulrich Schulmeister
Robert Bosch GmbH, Corporate Sector Research and Advance Engineering,
Robert-Bosch-Strasse 2, 71701 Schwieberdingen, Germany
christian.bertsch@de.bosch.com elmar.ahle@de.bosch.com ulrich.schulmeister@de.bosch.com

The demand for model exchange between development partners will grow during the next years. The Functional Mockup Interface (FMI) [1] is a well received tool independent approach for model exchange. The Original Equipment Manufacturers (OEM) have committed themselves to support FMI as exchange format for simulation models. Therefore, the FMI is a promising candidate to become the industry standard for model exchange and cross-company collaboration.

There are two complementary approaches for modeling complex systems:

- *White box modeling*: Modeling the entire system with one modeling language.
- *Black box model exchange*: Defining an interface for model exchange for standardized, tool-independent exchange format for simulation.

In this paper, the black box model exchange with FMI is evaluated from an industrial perspective. Requirements on such an interface for industrial applications are standardization of the model interface, availability of a significant number of supporting tools, easy-of-use of the interface, adoption of the standard, accompanying documentation and the maturity of such an interface. The paper focuses on the maturity of the FMI standard or the maturity of the implementation respectively.

In the MODELISAR project, requirements for FMI were derived from the beginning and tested for industrial applications. The Performance of the FMI approach was demonstrated in 24 industrial applications. In 2012, an internal benchmark at Bosch with three exporting tools and five importing tools showed quite different results. The test examples range from a "model" containing a sine generator only, a bouncing ball, a spring-damper system, an RC circuit to a thermal network. While some combinations worked quite well, other combinations did not work at all. The challenges encountered in the benchmark are classified and ordered with increasing maturity of the FMI standard. The maturity issues of FMI-based simulation were addressed to the FMI community at the MODELICA/FMI meetings beginning early 2012. The discussions and input from other companies and users resulted in the call for quotation of an FMU Compliance Checker and later of FMI Cross Checking rules. In the future, FMI Cross Checking should be extended to more tools and for multiple FMUs. The (co-)simulation techniques for importing tools should be improved and the improvements of FMI standard 2.0 should be implemented soon.

While FMI as a technical standard is right on track, there are other points to be addressed such as model exchange process and accompanying documentation. FMI is the best available approach for tool-independent model exchange and co-simulation. FMU Compliance Checking and FMI Cross Checking address technical problems that existed in the past. FMI Cross checking should be extended to more complicated examples. Then, FMI can become the technical basis for model-based collaborative engineering in a heterogeneous tool environment with different partners.

References

[1] Blochwitz, T. et al., The Functional Mockup Interface for Tool independent Exchange of Simulation Models, In: Proceedings of the 8th International Modelica Conference 2011, Dresden, Germany

An FMI-Based Tool for Robust Design of Dynamical Systems

Maria Henningsson, Johan Åkesson, Hubertus Tummescheit
Modelon AB, Modelon Inc
{maria.henningsson,johan.akesson,hubertus.tummescheit}@modelon.com

Model-based engineering is a key technology for competitive product development. However, implementing, parameterizing, and validating simulation models of physical systems is time-consuming and costly. To make modeling efforts pay off, it is necessary to systematically consider tools, practices, and workflows to get the most use out of a model portfolio.

Concepts from quality sciences, such as robust design, six-sigma, and design-of-experiments have had a large impact on product development in industry. These concepts are increasingly used in a model-based engineering context where data is gathered from simulation models rather than laboratory setups or prototypes.

This paper presents a framework to apply such ideas to analysis of dynamical systems. A set of tools that can be used for uncertainty analysis of dynamical Modelica models is presented. These tools are made available in the FMI Toolbox for MATLAB. The workflow and tools are demonstrated on a cooling loop design problem.

Simulating Rhapsody SysML Blocks in Hybrid Models with FMI

Yishai A. Feldman[1] Lev Greenberg[1] Eldad Palachi[2]
[1]IBM Research – Haifa, Israel [2]IBM Rational, Rehovot, Israel
{yishai,levg,eldadpal}@il.ibm.com

The Functional Mockup Interface (FMI) [1] standard enables hybrid simulation of models from different tools. Such tools can have different underlying behavioral semantics, creating challenges when models are combined. A case in point is the combination of the Rhapsody® tool [3, 2], widely used to describe and implement discrete control behavior in SysML, and Modelica®, widely used to model multi-domain physical systems.

This paper describes a plugin we developed for exporting Functional Mockup Units (FMUs) from Rhapsody, and the results of combining generated FMUs with models from other tools. The plugin is demonstrated in action through an example of a hybrid model, in which a controller specified as a SysML statechart is exported as an FMU and imported into a Modelica model in the SimulationX® tool [4].

When a Rhapsody FMU is used in a different environment, some basic assumptions on its behavior are challenged. As a result, modelers need to be aware of the fact that their models will be exported as FMUs and follow the guidelines described in the paper in order to ensure for the exported FMU to express their intentions.

For example, there are constraints on how continuous input variables can be used in Rhapsody; they must not be used to trigger Rhapsody events, otherwise they will cause thrashing of the simulation. Another issue is the fact that events in Rhapsody are sent asynchronously, while FMI semantics is synchronous. There are various strategies for the generated FMU to choose how many events are reported at each step, and these effect the semantics of the communication of the FMU with other units; the paper discusses several strategies and their implications.

Other issues include the fact that simulation time is counted in different units in FMI and in Rhapsody; this can also cause unexpected behavior of the FMU. Similarly, Rhapsody models can employ the full type system of the target language, while FMI defines a restricted set of types. The export plugin will choose the best FMI type corresponding to the Rhapsody type, to the extent possible.

Since different tools come with their own semantics, we expect that such mismatches are common, especially when connecting continuous-time with discrete models, and that our guidelines will generalize to many such cases.

References

[1] Functional Mockup Interface. `https://www.fmi-standard.org`.

[2] D. Harel and H. Kugler. The Rhapsody semantics of statecharts (or, on the executable core of the UML). In *Integration of Software Specification Techniques for Applications in Engineering*, pages 325–354. Springer, 2004.

[3] Rhapsody. `http://www-03.ibm.com/software/products/en/ratirhapfami`.

[4] SimulationX. `http://www.simulationx.com`.

Nonlinear State Estimation
with an Extended FMI 2.0 Co-Simulation Interface

Jonathan Brembeck[1], Andreas Pfeiffer[1], Michael Fleps-Dezasse[1],
Martin Otter[1], Karl Wernersson[2], Hilding Elmqvist[2]

[1]German Aerospace Center (DLR), Institute of System Dynamics and Control,
82234 Weßling, Germany

[2]Dassault Systèmes AB, Ideon Science Park, 22370 Lund, Sweden
Jonathan.Brembeck@dlr.de, Andreas.Pfeiffer@dlr.de, Michael.Fleps-Dezasse@dlr.de,
Martin.Otter@dlr.de, Karl.Wernersson@3ds.com, Hilding.Elmqvist@3ds.com

Abstract

In this paper we propose a method how to automatically utilize continuous-time *Modelica* models directly in nonlinear state estimators (Figure 1). The approach is based on an extended *FMI 2.0 Co-Simulation Interface* that interacts with the state estimation algorithms implemented in a *Modelica* library (Figure 2). Besides a short introduction to *Kalman Filter* based state estimation, we give details on a generic interface to cooperate with FMUs in *Modelica*, an implementation of nonlinear state estimation based on this interface, and the *Dymola* prototype used for the evaluation. Finally we show first results in a tire load estimation application (Figure 4) for DLR's robotic electric research platform *ROMO* (Figure 3).

Keywords: FMI 2.0 Co-Simulation, FMU, Event Handling, Inline Integration, Kalman Filter, State Estimation, Moving Horizon Estimation, Tire Load Estimation

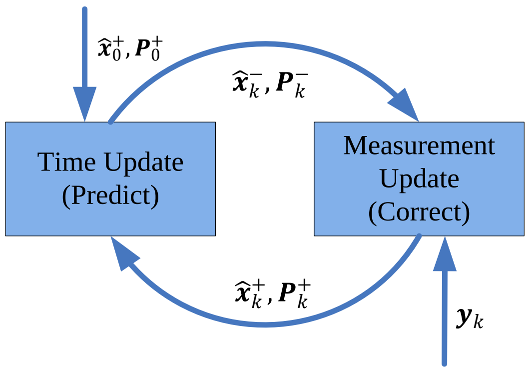


Figure 1: Principle of Kalman Filter based Estimation

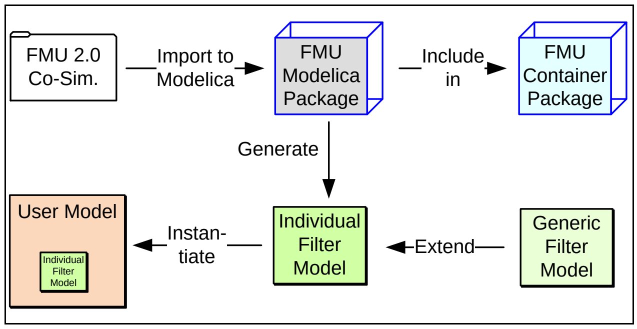


Figure 2:Process flow to generate a state estimator based on an FMU

Figure 3: ROMO on the four post test rig

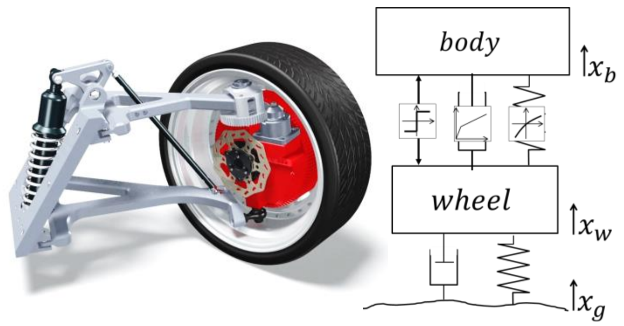


Figure 4:left: the "Wheel Robot" concept, right: nonlinear two mass system

Model-based Development of Future Small EVs using Modelica

Yutaka Hirano Shintaro Inoue Junya Ota
Toyota Motor Corporation, Future Project Division
1200 Mishuku, Susono, Shizuoka, 410-1193 JAPAN
{yutaka_hirano, shintaro_inoue_aa, junya_ota}@mail.toyota.co.jp

To cope with demands for future low carbon society, development of new-type small electric vehicles (EVs) becomes very active. To reduce the energy consumption in various actual driving conditions, considering overall running resistance such as aerodynamic resistance, tire rolling resistance including cornering drag, mechanical and electrical losses, etc. will be necessary. On the other hand, to cope with reduced stability against external disturbances such as side wind because of the light weight, it was clarified that additional control of direct yaw moment is effective. In this paper, model-based development of a new electric vehicle using Modelica is described. Full vehicle model considering both vehicle dynamics and energy consumption was developed as shown in Figure 1 and utilized to investigate the best possible solutions for both basic design of the vehicle and design of the control system.

There are three points in this paper. First, the influence of tire rolling resistance coefficient (RRC) and tire cornering power (CP) and also the vehicle weight for overall driving resistance and energy consumption was investigated. It became clear that light vehicle weight, low RRC value and high CP value are effective to reduce the overall driving resistance and thus energy consumption as Figure 2.

Secondly, it was also proved that light vehicle weight tends to result in reduced vehicle stability against external disturbances such as side wind. To cope with this problem, a new direct yaw moment control using a torque vectoring differential gear (TVD) was proposed. Also detailed investigation about dynamic performance and energy consumption of the new TVD systems was done.

Third point is that capability of recuperating breaking energy is also essential to improve the energy consumption. To clarify the necessary specification of the electric power system, the estimation of driving and breaking powers in various driving conditions were performed. Finally estimation of proper battery capacity for breaking power regeneration for various driving conditions was done.

It was evident that Modelica was very powerful to perform this kind of multi-physics and multi-discipline investigations for the holistic vehicle systems and control design in the early phase of development planning.

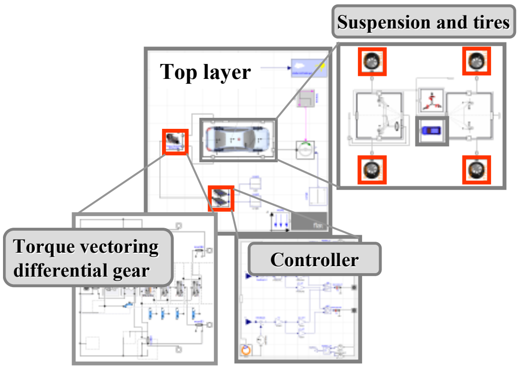


Figure 1: Total vehicle model

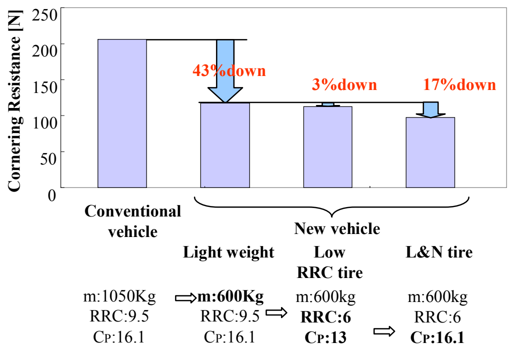


Figure 2: Comparison of cornering resistance force for various design condition

Modeling of an Electric Axle Drive with Modelica: A Study of Electric Active Dynamics

Hasan Flaih Awad Frank Rettig Tomas Smetana
Schaeffler Technology AG & CO. KG
Industriestrasse 1-3, 91074 Herzogenaurach, Germany
{awadhsa, rettifan, smetatma}@schaeffler.com

This paper focuses on modeling of the electric axle drive system (eAxle) [Figure 1] used for improving vehicle stability and handling performance by means of a torque vectoring (TV) feature as well as improving vehicle traction and reducing CO2 emissions by means of an electric traction feature. The function, construction and benefits of the eAxle will be explained within these contexts. An overview of the modeling of the eAxle in Dymola® will be shown. Several simulation cases are conducted to verify the effectiveness of the system for reducing fuel consumption and improving longitudinal and lateral dynamics. A co-simulation was developed between Dymola® and Abaqus® to simulate the power loss and ascertain the temperature behavior on the housing of the eAxle . Finally, the eAxle with a vehicle model was driven over a special realistic handling course using an open source software called Blender®. The dynamic behavior of the whole vehicle model (with eAxle) will be validated by means of an optical measurement process or what so called Object-Tracking.

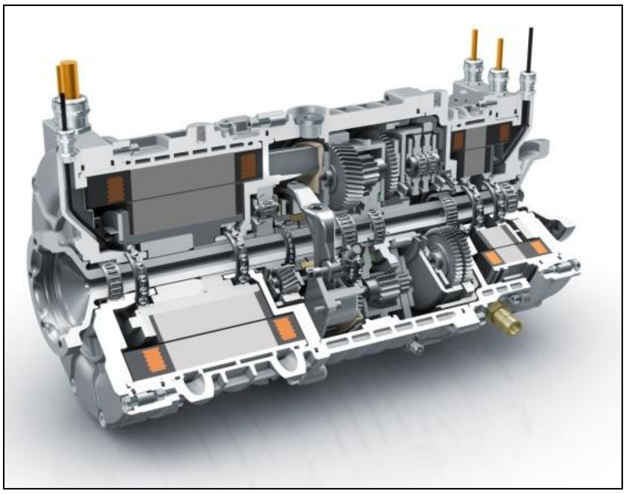


Figure 1:High voltage electric axle drive

References

[1] „Schaeffler E-Axle Drive. Dr. T. Smetana

[2] „Lastdatenermittlung in Elektrischen Achsantrieben und Betriebsfestigkeitsbewertung für das Subsystem Elektrische-Maschine".
D. Knetsch, F. Rettig, Dr. M. Funk, H. Awad, Dr. T. Smetana, M. Gramann. 40.Tagung des DVM-Arbeitskreises Betriebsfestigkeit.

Utilizing Object-Oriented Modeling Techniques for Composition of Operational Strategies for Electrified Vehicles

Sebastian Hämmerle
Vorarlberg Univ. of Appl. Sc.
Austria
Sebastian.Haemmerle@students.fhv.at

Marco Kessler
Modelon GmbH
Germany
Marco.Kessler@modelon.com

Markus Andres
Modelon GmbH
Germany
Markus.Andres@modelon.com

The paper introduces a new concept of modeling the overall control unit of hybrid electric vehicles in *Modelica*. The work focuses on a structure which can simulate substantially different vehicle concepts without changing the structure of the control unit. Based on this universal implementation different scenarios can be simulated rapidly and consequently cheaply, including fundamentally different drive trains ranging from conventional to purely electrical including hybrid versions. The paper focuses on components that are responsible for the generation of traction force.

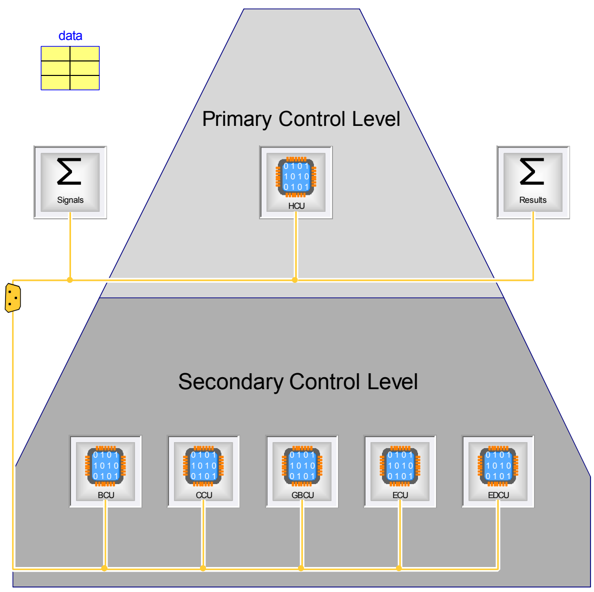


Figure 1: Template of the universal CU

The modeled control unit (Figure 1) is designed for one-dimensional simulations, therefore taking accelerator and brake pedal positions computed by a separated driver model as inputs. Based on these inputs and the vehicle's internal states the HCU's primary control level decides which one out of eight driving modes is activated. Based on the driving mode, the component controllers in the secondary control level compute input signals for the corresponding component.

The full paper demonstrates that simulations of purely electrical, conventional and hybrid vehicles can be carried out with minimal changes in the controller's parameters. The vehicles of choice include BMW's recent i-Series lineup, namely the purely electrical i3 (alternatively with range extender) and the through-the-road hybrid i8 representing a very complex drive-line structure.

Thermal shock testing for Engines in Dymola

Alessandro Picarelli† Eduardo Galindo* Gonzalo Diaz*

†*Claytex Services Ltd.,*
Edmund House, Rugby Road, Leamington Spa, CV32 6EL, United Kingdom

**AVL IBERICA SA - OFICINA VALLADOLID*
P. Tecn. Castilla-León, Edif. Centro, E-47151, Boecillo (Valladolid)

†alessandro.picarelli@claytex.com *Eduardo.Galindo@avl.com
*Gonzalo.Diaz@avl.com

In this work, we use an acausal multi-domain physical system model to study the interaction between an internal combustion engine operation and a range of cooling scenarios. Although the model can be used for modelling a wide range of scenarios, in this paper we concentrate on the application of “thermal shock”.

Many manufacturers carry out thermal shock tests to understand and prevent component failure, as well as to accelerate durability testing of engines and engine components, including cylinder-head gaskets. An internal combustion engine is load-controlled on a dynamometer and coolant temperature transients are imposed on the engine system. Using freely available and commercial Modelica Libraries within the Dymola environment, the whole system integration of the coolant rig and engine dynamometer is achieved. This allows the user to develop and define control strategies for the tests from desktop, prior to engaging in the real tests.

The commercial library used to model the internal combustion engine is the Claytex Engines library [1]. This library is capable of modelling both mean value and crank angle resolved engine models and is fully compatible with the Modelica Standard library. The Modelica Standard library forms the basis for the cooling circuit modelling.

The engine being tested is a 4 cylinder inline Turbo Diesel engine with direct injection. The mean value version model of the engine is installed in a torque controlled dynamometer and coupled to the cooling rig.

The integrated systems including the engine, cooling rig and associated controllers are run within the Dymola environment to predict the engine heat rejection and cooling rig performance in a thermal shock scenario. A good match of experimental data vs. simulation results is achieved for the thermal shock coolant temperatures. The models perform faster than real time with both variable and fixed step solvers. The fixed step solver simulation with inline integration is shown to be capable of meeting and exceeding the required 250Hz sampling for HIL (Hardware In the Loop) simulation.

Keywords: Engine testing, thermal-shock, control system development

References

[1] Dempsey M., and Picarelli A. (2009). Investigating the multibody dynamics of the complete powertrain system. Como, Italy: Proceedings 7th Modelica Conference.

Model-Based Design of Integrative Energy Concepts for Building Quarters using Modelica

Dipl.-Ing. Torsten Schwan Dipl.-Ing. René Unger
EA Systems Dresden GmbH, Königstr. 2, 01097 Dresden, Germany
torsten.schwan@ea-energie.de rene.unger@ea-energie.de

Dr.-Ing. Christian Lerche
Ingenieurbüro Dr. Lerche, Lugstr. 5, 01796 Pirna, Germany
doclerche@web.de

Dipl.-Ing. Christian Kehrer
ITI GmbH, Schweriner Str. 1, 01067 Dresden, Germany
kehrer@itisim.com

Increasing energy prices as well as outdated building systems present the housing industry with the challenge of finding new complex system solutions including renewable energy and storage systems. The municipality Lohmen (Germany) and the local housing association contracted EA Systems and IB Dr. Lerche to develop an integrative energy system concept for its historic town center.

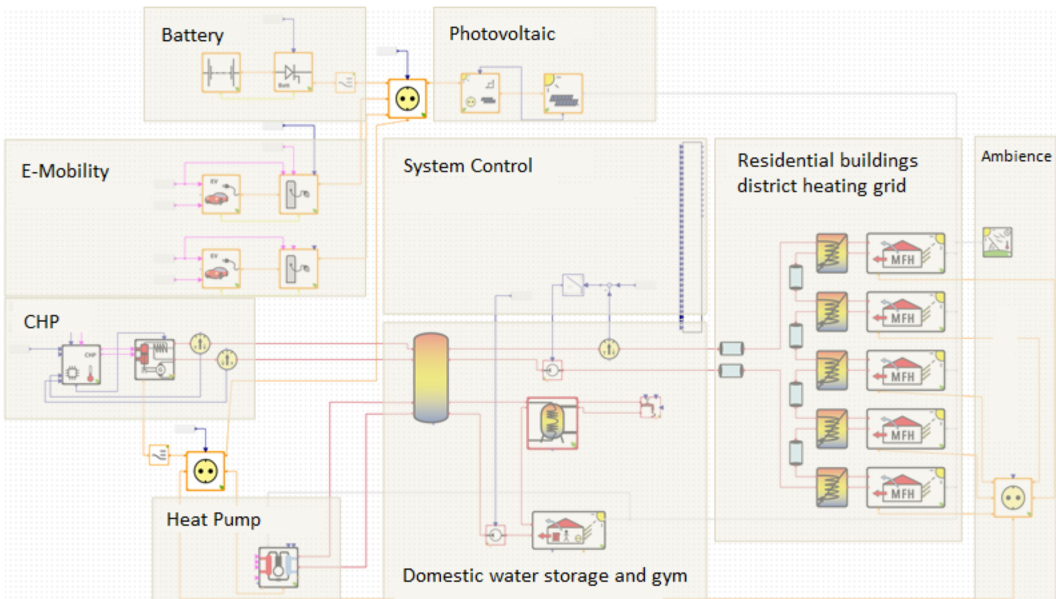


Figure 1: Modelica-based simulation model of renewable energy system concept for historic town center in Lohmen (Germany)

This paper deals with modeling and simulating different energy system variants for the existing building structure using the Modelica-based ‘Green Building’ library and SimulationX. The discussion illustrates the challenges of the modeling process, innovative solutions and the simulation results.

Enhancement of the building simulation software TRNSYS by coupling to the VEPZO model programmed in Modelica

Svea Kübler[1,2] Victor Norrefeldt[1]
1: Fraunhofer-Institute for Building Physics, Fraunhoferstr. 10, D-83626 Valley
2: Transsolar Energietechnik GmbH, Curiestr. 2, D-70563 Stuttgart

svea.kuebler@ibp.fraunhofer.de victor.norrefeldt@ibp.fraunhofer.de

In this study, the possibility to interface a commercial building simulation tool with Modelica models is investigated. In this application, the zonal model VEPZO – modeled in Modelica – is coupled to the software TRNSYS – mainly programmed in Fortran – to be able to perform a dynamic co-simulation. The objective of this coupling is to obtain refined airflow and air temperature prediction, while retaining computation effort low enough to allow for transient computation. In a first attempt, a coupling using FMI was tested without success due to a lack of adequate solvers for FMI export. Therefore, a script coupling was implemented. Further steps include a validation and evaluation of the programmed interface and the results of the coupled system in respect to computation time, quality of results, usability and further development.

The Modelica Thermal Model Generation Tool for Automated Creation of a Coupled Airflow, Radiation Model and Wall Model in Modelica

Arnav Pathak, Victor Norrefeldt, Abdellah Lemouedda, Gunnar Grün

Fraunhofer Institute for Building Physics, Dept. Indoor Climate, 83626 Valley, Germany

arnav.pathak@ibp.fraunhofer.de, victor.norrefeldt@ibp.fraunhofer.de, abdellah.lemouedda@ibp.fraunhofer.de , gunnar.gruen@ibp.fraunhofer.de

This paper presents the Modelica Thermal Model Generation Tool. The aim of this tool is to enable the user to set up a geometrically correct thermal model for complex geometries that allows predicting the impact of heated/heating devices and their location both in terms of airflow pattern and radiation distribution. Using a geometry file exported from CAD software, the tool distributes wall facets, air nodes and computes the long-wave radiant view factor matrix for obstructed and unobstructed surfaces. This information is exported as ready to use Modelica code. The zonal model VEPZO is used to model airflow within a domain (enclosed space). This model allows predicting airflow and air temperature distribution in space on a coarse mesh and thus computes faster than classical CFD computations. Walls are subdivided on the same grid as the zonal model is set upon. For each wall facet, the Modelica Thermal Model Generation Tool computes the view factors to the other facets in the domain.

Comparison of simulated results with test data and application of the Modelica Thermal Model Generation Tool for a room with radiant heating and for the cooling of an aircraft cockpit are presented in this paper.

Modelling long-wave radiation heat exchange for thermal network building simulations at urban scale using Modelica

Moritz Lauster, Peter Remmen, Marcus Fuchs, Jens Teichmann,
Rita Streblow, Dirk Müller
RWTH Aachen University, E.ON Energy Research Center, Institute for Energy Efficient Buildings
Mathieustr. 10, 52074 Aachen, Germany
mlauster@eonerc.rwth-aachen.de

There are different options for modelling indoor and outdoor long-wave radiation exchange in thermal building models for simulations at urban scale. For improving these building models, a good trade-off between accuracy and simulation time is a major challenge.

During investigations of individual building model components with the help of the American Standard ASHRAE 140 [1] , we identified long-wave radiation exchange as one key part for optimization. It seems to have major influence on heat demand [2]. This paper provides four different methods to calculate outdoor long-wave radiation exchange. While three of them are based on the implementation of an equivalent air temperature, one calculates the heat exchange based on the Stefan-Boltzmann law. In addition, two methods to calculate indoor long-wave radiation exchanges are tested, one based on a linear approach, another based on Stefan-Boltzmann law.

For the comparison, we set-up three test cases on a generic room and a single family dwelling and analysed surface temperatures, heat demands and simulation times. The generic room corresponds to a test case of ASHRAE 140; the single family dwelling was constructed in accordance to German Energy Savings Ordinance 2009 with a high thermal mass. We varied the number of radiation sources between the test cases to observe radiation heat exchange under generic and real conditions.

The results of the test cases show promising potential for an outdoor radiation exchange model based on a modified approach from German Guideline VDI 6007. It includes important simplifications that lead to short computing time while keeping a sufficient accuracy. For indoor radiation exchange modelling at constant temperatures, a linear approach significantly reduces simulation time without any major accuracy losses.

Modelica proved thereby to be a promising modelling language for urban scale building simulations. Major prerequisites and advantages are solvers with variable time steps and the use of object-oriented and acausal modelling approaches.

References

[1] American Society of Heating, Refrigerating and Air-Conditioning Engineers. Standard Method of Test for the Evaluation of Building Energy Analysis Computer Programs(ASHRAE 140-2007). Atlanta: ANSI/ASHRAE; 2007.

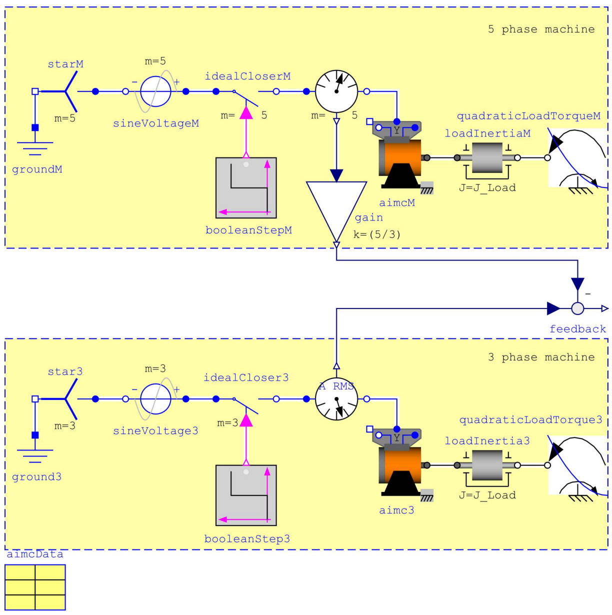
[2] Shi Z, Zhang X. Analyzing the effect of the longwave emissivity and solar reflectance of building envelopes on energy-saving in buildings in various climates. Solar Energy 2011;85(1):28–37.

Extension of the FundamentalWave Library towards Multi Phase Electric Machine Models

Christian Kral, Electric Machines, Drives and Systems, 1060 Vienna, Austria
Anton Haumer, Technical Consulting, 3423 St.Andrä-Wördern, Austria
Reinhard Wöhrnschimmel, AIT GmbH, 1210 Vienna, Austria

dr.christian.kral@gmail.com, a.haumer@haumer.at

Electric machine theory and electric machine simulations models are often limited to three phases. Up to the Modelica Standard Libray (MSL) version 3.2 the provided machine models were limited to three phases. Particularly for large industrial drives and for redundancy reasons in electric vehicles and aircrafts multi phase electric machines are demanded. In the MSL 3.2.1 an extension of the existing FundamentalWave library has been performed to cope with phase numbers greater than or equal to three. The developed machine models are fully incorporating the multi phase electric, magnetic, rotational and thermal domain. In this publication the theoretical background of the machines models, Modelica implementation details, the parametrization of the models and simulation examples are presented.



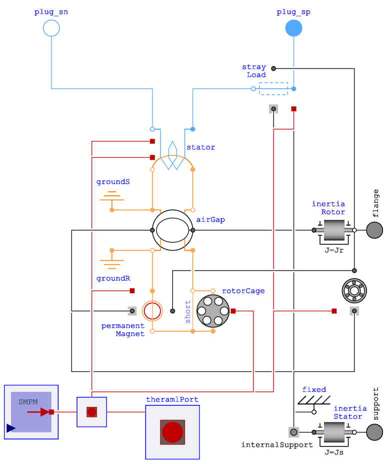
New Multi Phase Quasi Static Fundamental Wave Electric Machine Models for High Performance Simulations

Christian Kral, Electric Machines, Drives and Systems, 1060 Vienna, Austria
Anton Haumer, Technical Consulting, 3423 St.Andrä-Wördern, Austria

dr.christian.kral@gmail.com, a.haumer@haumer.at

A new quasi static fundamental wave machines library will be included in the magnetic domain package of the next Modelica Standard Library (MSL). The provided classes of machine models omit all transient electrical effects, but mechanical dynamics are fully taken into account. By including the new machine models new classes of problems can be treated, enabling fast electric machine and drive simulations. Yet, all the characteristic loss effects of transient machine models are fully taken into account, where needed. Phase numbers greater than three are supported. For each machine type available in the MSL there will then exist both a fully transient and a quasi static electric machine model.

The package structure of the quasi static fundamental wave package and the concept of implementation will be presented. All required assumptions and limitations for operating the new machine models will be presented and discussed. Deviating parameters compared to the transient machine models will be discussed and explained. Simulation examples will be presented and compared with transient simulation experiments. Possible applications for the new machine models will be outlined.



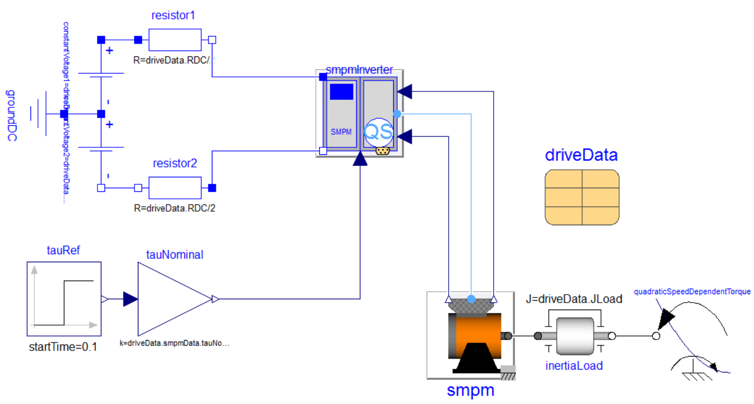
The New EDrives Library: A Modular Tool for Engineering of Electric Drives

Anton Haumer, Technical Consulting, 3423 St.Andrä-Wördern, Austria
Christian Kral, Electric Machines, Drives and Systems, 1060 Vienna, Austria

www.edrives.eu

Abstract

Simulation is an indispensable tool for the engineering of systems containing electric drives. Depending on the design phase and the engineering task different levels of modeling details are required: proof of concept, investigation of energy and power consumption, design of control, etc. The new EDrives library provides three levels of abstraction for inverters: quasi static (neglecting electrical transients), averaging (neglecting switching effects) and switching – for serving different demands. The inverters can feed the machine models of the Modelica Standard Library:
Modelica.Magnetic.FundamentalWave and the new
Modelica.Magnetic.QuasiStatic.FundamentalWave. The EDrives library copes with arbitrary phase numbers and can be easily extended to develop new control algorithms. In this publication the structure of the library and the implemented control principles are presented. Furthermore, examples comparing the three different levels of abstraction are included.



Modelica Models for Magnetic Hysteresis, Materials and Transformers

Johannes Ziske Thomas Bödrich
Technische Universität Dresden, 01062 Dresden, Germany
{Johannes.Ziske, Thomas.Boedrich}@tu-dresden.de

This paper presents new extensions of the Modelica.Magnetic.FluxTubes library [1] which will be integrated into the Modelica Standard Library soon. The extensions mainly are hysteresis elements for modeling of ferromagnetic and dynamic hysteresis of magnetic materials during transient simulation of electromagnetic components with lumped network models. Two different static hysteresis models, the Tellinen [2] and the Preisach model [3] have been implemented. Dynamic hysteresis is accounted for with a d*B*/d*t* term. Consideration of both the static and dynamic hysteresis during transient simulation allows, among others, for determination of hysteresis losses. This becomes more and more important during design of electromagnetic actuators and systems due to increasing requirements on power density and miniaturization. A new material package, mainly based on in-house measurements, provides hysteresis data of several magnetic materials for a one-click configuration of the hysteresis elements. Additionally, based on the hysteresis elements, models for permanent magnets as well as single- and three-phase transformer models have been implemented. An exemplary model of a single-phase transformer including simulation results is shown in Figure 1.

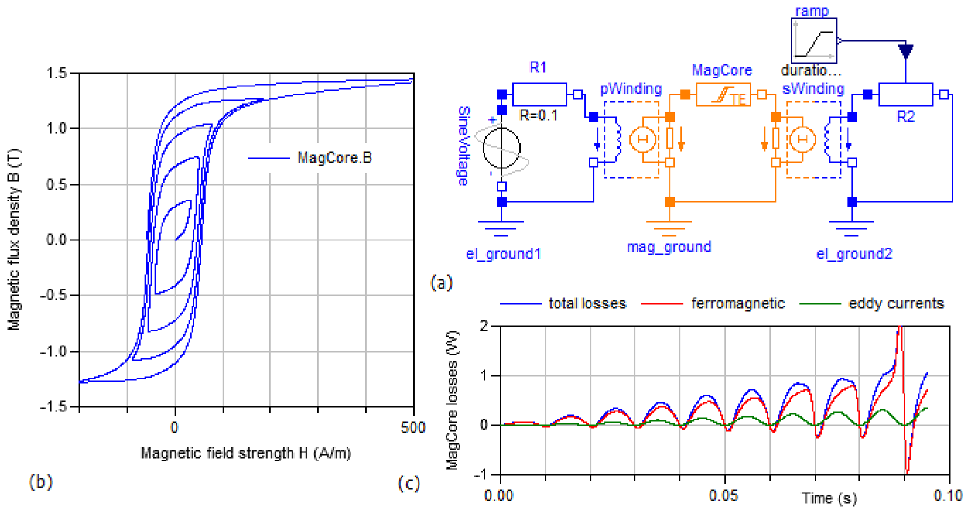


Figure 1: Model of a single-phase transformer with the new hysteresis element MagCore and a ramped load (a), resulting *B*(*H*) hysteresis loops of the magnetic core (b) and the corresponding hysteresis losses divided into their ferromagnetic and eddy current components (c)

References

[1] T. Bödrich: Electromagnetic Actuator Modelling with the Extended Modelica Magnetic Library, Proc. of 6th Int. Modelica Conf., Bielefeld, Germany, March 3-4, pp. 221–227, 2008

[2] J. Tellinen: A Simple Scalar Model for Magnetic Hysteresis, IEEE Transactions on Magnetics, vol. 24, no. 4, pp. 2200 – 2206, July 1998

[3] F. Preisach: Über die magnetische Nachwirkung, Zeitschrift für Physik A Hadrons and Nuclei, vol. 94, pp. 277–302, 1935

Custom Annotations: Handling Meta-Information in Modelica

Dirk Zimmer[1], Martin Otter[1], Hilding Elmqvist[2], Gerd Kurzbach[3]
[1]German Aerospace Center (DLR), Institute of System Dynamics and Control, D-82234 Wessling, Germany,
[2]Dassault Systèmes AB, Ideon Research Park, SE-223 70 Lund, Sweden
[3]ITI GmbH, 01067 Dresden, Germany
dirk.zimmer@dlr.de, martin.otter@dlr.de, hilding.elmqvist@3ds.com, kurzbach@itisim.com

Annotations and attributes form an important part of the Modelica language. They are used to include various meta-information such as documentation, external C-code, compilation hints, etc. Given the increasingly wide field of potential applications the set of useful annotations becomes too large to be included in the language specification. Hence we present a proposal how a Modelica modeler may define his own annotations and how such custom annotations can be organized within Modelica libraries. In the long term, the goal is to move the definition of standardized annotation, as well as of attributes, from the Modelica specification to a standard library.

For example, a parameter can be marked to be a "Tuner" that shall be optimized. The following statement

```
parameter Real p1 annotation(OptimSetup.Tuner(active=true, min=-2));
```

indicates that for a default optimization setup for this model, parameter `p1` shall be used as tuner and shall have a constraint `p1 >= -2`. A custom annotation is identified by the full path name of the Tuner record class. This defines a new instance of the record, together with a modifier on this record. So, conceptually, this custom annotation is equivalent to the following declaration:

```
OptimSetup.Tuner name(active=true, min=-2);
```

and the name of the instance is not defined, because not needed (the identification of the data is via the class name).

It is proposed to optionally store custom annotations in the `modelDescription.xml` file when exporting the model as Functional Mockup Unit (FMU). Since custom annotation variables are basically standard Modelica variables with all the attributes of Modelica variables, it is proposed to just define them as standard FMI variables and mark the "custom annotation" property in the name. In particular, the name of a custom annotation variable shall be:

```
<ComponentName>.annotation.<CustomAnnotationFullClassName>.<elementName>
```

With a script language the xml-file can be read, the custom annotations retrieved and used to build automatically a default setup for special analysis or synthesis environments.

The proposed concept has been evaluated with a Dymola prototype.

Modelica extensions for Multi-Mode DAE Systems

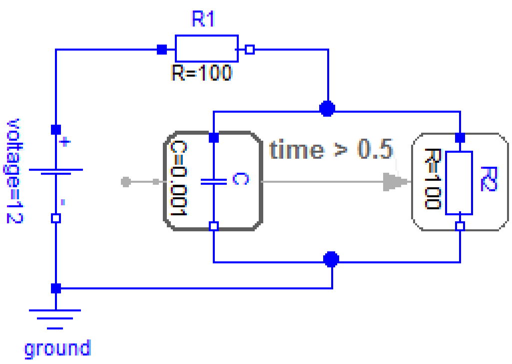
Hilding Elmqvist[1], Sven Erik Mattsson[1], Martin Otter[2]
[1]Dassault Systèmes AB, Ideon Science Park, SE-223 70 Lund, Sweden
[2]German Aerospace Center (DLR), Institute of System Dynamics and Control,
D-82234 Wessling, Germany,
Hilding.Elmqvist@3ds.com, SvenErik.Mattsson@3ds.com, Martin.Otter@dlr.de

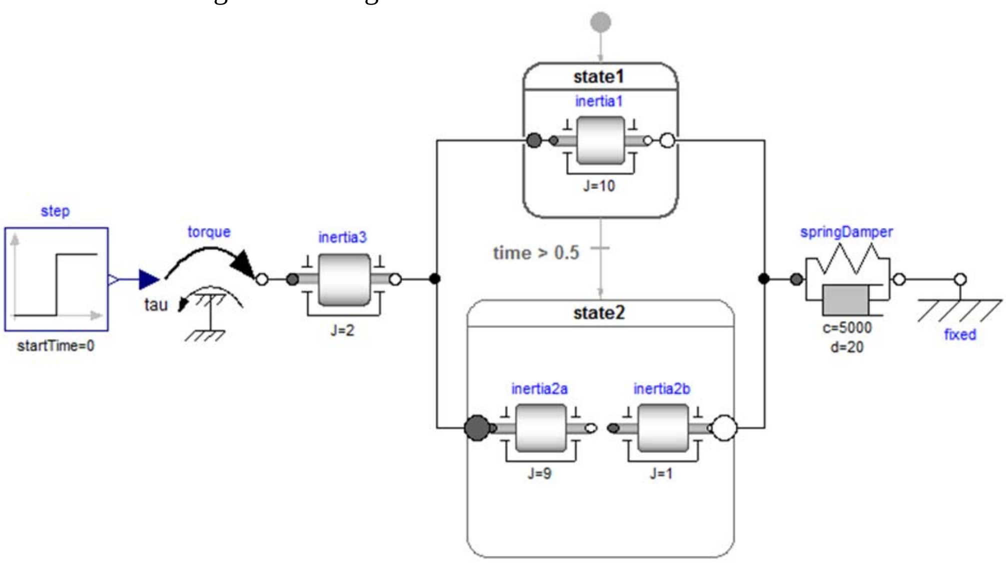
The paper proposes to extend the scope of Modelica and model and simulate systems with multiple operating modes. Examples:

- Changing a controller from nominal operation to startup, shutdown or manual operation.
- Structural changes of a physical model (e.g. modeling the separation mechanism of a two or three stage rocket).
- Describing failure situations where the model structure is changing (e.g. an electrical line or a mechanical shaft breaks).
- Describing failure situations where the model is completely changing (e.g. the normal behavior of a pump is a 0D model. In case of cavitation, a 1D model is needed to describe physics, requiring to switch dynamically from a 0D to a 1D model when cavitation occurs).

In general this means that the number of continuous-time states of the model might change dynamically during the simulation. Such models cannot be described with current Modelica, version 3.3, since in this case the basic requirement is that the number of continuous-time states of a model is fixed during a simulation.

The basic idea for multi-mode modeling is to extend the Modelica 3.3 synchronous state machines to continuous-time state machines having continuous-time models as “states”. Every model can be a “state” of a state machine, including acausal models, such as a capacitor, or a rotational inertia. The number of continuous-time states can change at a transition of a state machine. No new Modelica language element is needed, but a generalized semantics for state machines has to be introduced, such as “merge semantics for differential equations“. The concepts have been evaluated with a Dymola prototype. Examples that can be handled with this prototype are shown in the figure to the right and below.

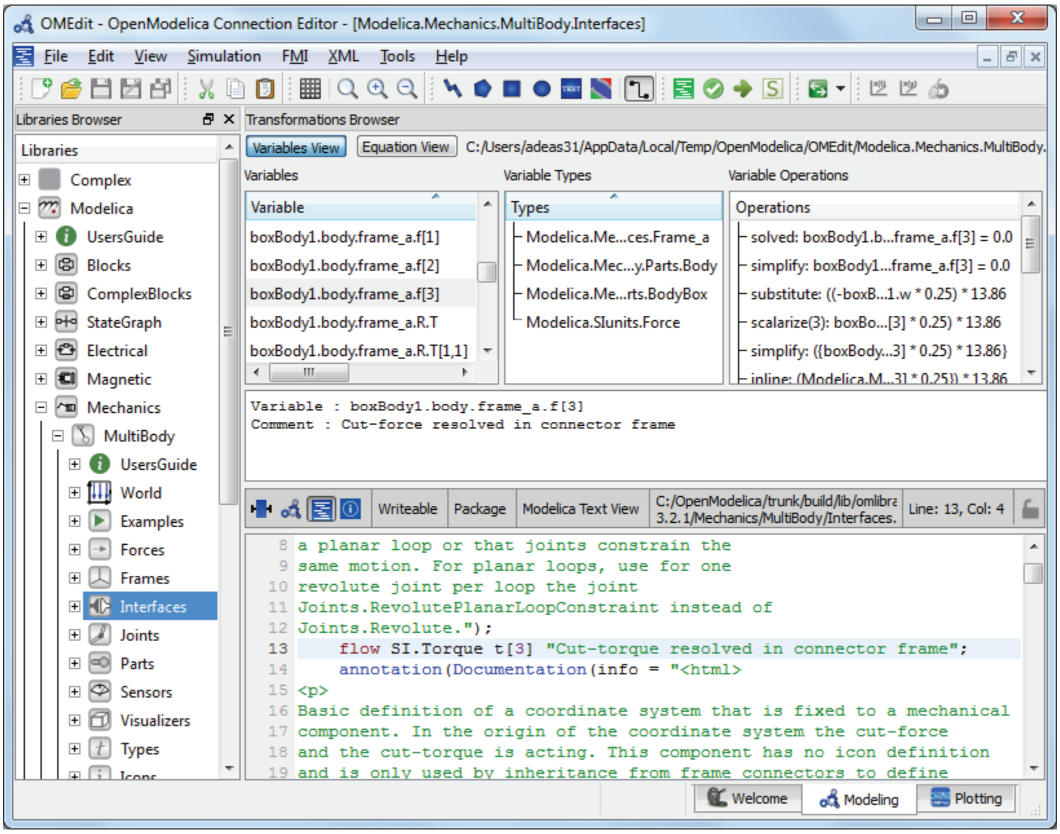




Integrated Debugging of Equation-Based Models

Martin Sjölund[1], Francesco Casella[2], Adrian Pop[1], Adeel Asghar[1], Peter Fritzson[1],
Willi Braun[3], Lennart Ochel[3], Bernhard Bachmann[3]
[1]Programming Environments Laboratory
Department of Computer and Information Science
Linköping University, Linköping, Sweden
[2]Dipartimento di Elettronica e Informazione, Politecnico di Milano, Milano, Italy
[3] Dept. Mathematics and Engineering, University of Applied Sciences,
D-33609 Bielefeld, Germany
{adrian.pop,martin.sjolund,adeel.asghar,peter.fritzson}@liu.se
francesco.casella@elet.polimi.it,
{bernhard.bachmann,lennart.ochel,willi.braun}@fh-bielefeld.de

The high abstraction level of equation-based object-oriented languages (EOO) such as Modelica has the drawback that programming and modeling errors are often hard to find. In this paper we present the first integrated debugger for equation-based languages like Modelica, which can combine static and dynamic methods for run-time debugging of equation-based Modelica models during simulations. This builds on and extends previous results from a transformational static equation debugger and a dynamic debugger for the algorithmic subset of Modelica.



Making Modelica Applicable for Formal Methods

Matthew Klenk Daniel G. Bobrow Johan de Kleer Bill Janssen
Palo Alto Research Center
3333 Coyote Hill Rd, Palo Alto, CA, 94304
{klenk,dekleer,bobrow,janssen}@parc.com

Engineers need to perform many different types of analyses as they design systems. Modelica has become a leading language to support numerical simulation. As a consequence there is widespread understanding of Modelica and a large number of Modelica model libraries available. This paper addresses the task of analyzing Modelica models with formal methods to derive system properties such as whether a design meets its requirements for all possible inputs. We report on our experience building a qualitative reasoner operating on Modelica models. In particular, we discuss the importance of leveraging the Modelica compiler to construct models for verification.

Modelica's reuse and flexibility are central to its appeal among designers, engineers, and researchers. Unfortunately, these features can also impede the application of formal methods to Modelica models. In this paper, we highlight five classes of problematic Modelica modeling practices:

- Artifacts for numeric simulation
- Unnecessary component model complexity
- Procedures
- Sequential states
- Incomplete models

For each class of modeling practices, we discuss a number of examples highlighting different ways in which the problem manifests. In each case, we seek to answer the following questions:

- Why do designers use it?
- Why is this difficult for formal methods?
- What should be done to enable formal verification?

We close the paper with a discussion of modeling principles to guide modelers in the future.

Implementing stabilized co-simulation of strongly coupled systems using the Functional Mock-up Interface 2.0

Antoine Viel
LMS Imagine
7 Place des Minimes, 42300 Roanne, France
antoine.viel@lmsintl.com

This paper addresses the main issue encountered with the co-simulation of coupled systems that exchange energy, i.e. the trade-off between computational performances and numerical stability. The possible loss of stability is explained by the discretization of the coupling variables and can be analyzed by considering such co-simulated systems as closed-loop sampled-time systems, as shown on figure 1.

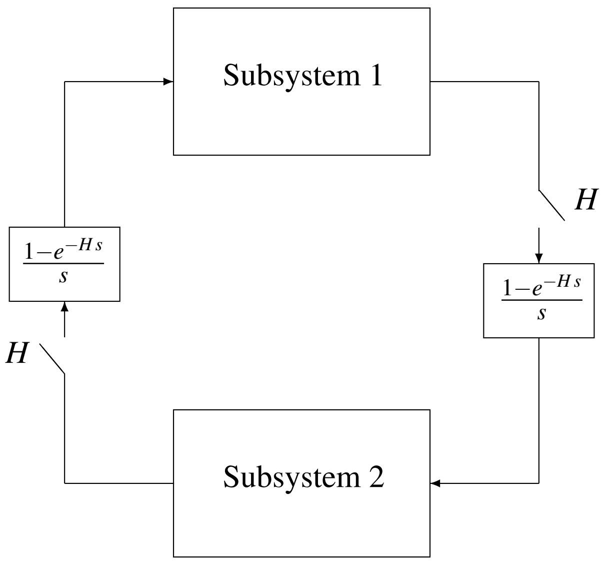


Figure 1: Bloc diagram of a two-subsystems co-simulation process.

This property is first explained in details with the help of a simple generic test system for which a large oversampling with respect to the Nyquist frequency is required in order to keep a good level of accuracy. The linearly implicit stabilization method from [1] is then implemented and tested thanks to the directional directives computation capability of the FMI for Co-simulation 2.0 standard [2]. Some minor extensions to the standard are proposed to efficiently implement the method. When applied to the test system, it is shown that large co-simulation steps can be taken, and hence significant computation time speed-ups are observed.

References

[1] Arnold M., Numerical stabilization of co-simulation techniques, the ODE case, Working document MODELISAR, sWP 200-203, September 5, 2011.

[2] Modelica Association Project "FMI", Functional Mock-up Interface for Model Exchange and Co-simulation 2.0 Release Candidate 1, October 18, 2013.

Context-based polynomial extrapolation and slackened synchronization for fast multi-core simulation using FMI

Abir Ben Khaled[1] Laurent Duval[1] Mongi Ben Gaid[1] Daniel Simon[2]

[1]IFP Energies nouvelles, 1 et 4 avenue de Bois-Préau,
92852 Rueil-Malmaison, France
[2]INRIA and LIRMM - DEMAR team, 95 rue de la Galéra,
34090 Montpellier, France
abir.ben-khaled@ifpen.fr laurent.duval@ifpen.fr
mongi.ben-gaid@ifpen.fr daniel.simon@inria.fr

The growing complexity of systems, together with increasing available parallelism provided by multi-core chips, calls for the parallelization of simulation. Simulation speed-ups are expected from co-simulation and parallelization based on models splitting into loosely coupled sub-systems in the framework of Functional Mockup Interface (FMI). However, slackened synchronization between the sub-models and associated solvers running in parallel introduces integration errors, which must be kept inside predefined bounds.

In this paper, context-based extrapolation is investigated to improve the trade-off between integration speed-ups, needing large communication steps, and simulation precision, needing frequent updates for the models inputs. An internal combustion engine, based on FMI for model exchange, is used to assess the parallelization methodology.

Keywords

FMI; parallel simulation; signal processing; polynomial extrapolation; real-time; context-based decision.

Model-Based Integration Platform for FMI Co-Simulation and Heterogeneous Simulations of Cyber-Physical Systems

Himanshu Neema[1] Jesse Gohl[2] Zsolt Lattmann[1] Janos Sztipanovits[1]
Gabor Karsai[1] Sandeep Neema[1] Ted Bapty[1] John Batteh[2]
Hubertus Tummescheit[2] Chandrasekar Sureshkumar[2]
[1]Institute for Software Integrated Systems, Vanderbilt University
1025 16th Avenue South, Suite 102, Nashville, TN 37212, USA
[2]Modelon, Inc.
2389 Main Street, Glastonbury, CT 06033, USA
himanshu@isis.vanderbilt.edu jesse.gohl@modelon.com lattmann@isis.vanderbilt.edu sztipaj@isis.vanderbilt.edu gabor@isis.vanderbilt.edu sandeep@isis.vanderbilt.edu bapty@isis.vanderbilt.edu john.batteh@modelon.com hubertus.tummescheit@modelon.com chandrasekar.sureshkumar@modelon.com

Virtual evaluation of complex Cyber-Physical Systems (CPS) [1] with a number of tightly integrated domains such as physical, mechanical, electrical, thermal, cyber, etc. demand the use of heterogeneous simulation environments. Our previous effort with C2 Wind Tunnel (C2WT) [2] [3] attempted to solve the challenges of evaluating these complex systems as-a-whole, by integrating multiple simulation platforms with varying semantics and integrating and managing different simulation models and their interactions. Recently, a great interest has developed to use Functional Mockup Interface (FMI) [4] for a variety of dynamics simulation packages, particularly in the automotive industry. Leveraging the C2WT effort on effective integration of different simulation engines with different Models of Computation (MoCs), we propose, in this paper, to use the proven methods of High-Level Architecture (HLA)-based model and system integration. We identify the challenges of integrating Functional Mockup Unit for Co-Simulation (FMU-CS) in general and via HLA [5] and present a novel model-based approach to rapidly synthesize an effective integration. The approach presented provides a unique opportunity to integrate readily available FMU-CS components with various specialized simulation packages to rapidly synthesize HLA-based integrated simulations for the overall composed Cyber-Physical Systems.

References

[1] J. Sztipanovits: Composition of cyber-physical systems, 14th Annual IEEE Int'l. Conference and Workshops on the Engineering of Computer-Based Systems (ECBS '07), Washington, DC, USA, IEEE Computer Society, 2007, pp. 3–6.

[2] Graham Hemingway, Himanshu Neema, Harmon Nine, Janos Sztipanovits, Gabor Karsai: Rapid Synthesis of High-Level Architecture-Based Heterogeneous Simulation: A Model-Based Integration Approach", Simulation 88(2), 217-232 (2012).

[3] C2WT community wiki: wiki.isis.vanderbilt.edu/OpenC2WT.

[4] Functional Mock-up Interface: www.fmi-standard.org.

[5] HLA standard – IEEE standard for modeling and simulation (M&S) high-level architecture (HLA) – framework and rules: ieeexplore.ieee.org/servlet/opac?punumber=7179.

Adapting Functional Mockup Units for HLA-compliant Distributed Simulation

Faruk Yılmaz, Umut Durak, Koray Taylan
Roketsan Missiles Inc.
Ankara, Turkey
[fyilmaz|udurak|ktaylan]@roketsan.com.tr

Halit Oğuztüzün
Middle East Technical University
Ankara, Turkey
oguztuzn@metu.edu.tr

ABSTRACT

Conceptual design of systems requires aggregate level simulations of the designed system in its operational setting. By this way, performance of the system and its interactions with the other entities in its environment can be evaluated. The complex nature of these simulations often requires distributed execution. IEEE 1516 High Level Architecture (HLA) is a widely accepted standard architecture for distributed aggregate level simulations. Functional Mock-up Interface (FMI) is a recent standardization effort that leads to a tool independent systems simulation interface that enables model reuse and co-simulation. This paper aims to present a method for developing HLA-compliant federates using FMI. The method enables a Functional Mock-up Unit to join an HLA-compliant federation as a member.

Keywords: Functional Mockup Interface; High Level Architecture; Distributed Simulation

Transmission Modeling in Modelica: A consistent approach for several software development platforms

Jochen Köhler, Michael Kübler, Julian King
ZF Friedrichshafen AG
Graf-von-Soden-Platz 1, D-88046 Friedrichshafen, Germany
{jochen.koehler, michael.kuebler, julian.king}@zf.com

Abstract

Simulation models play a fundamental role in the development of transmission control software. In the ideal case, the same model can be used throughout the whole development process from concept and design over implementation to system verification. The idea is to use one uniform model along this V-scheme. This leads to the requirement that simulations have to be able to run in real-time on hardware-in-the-loop platforms. On the other hand, very detailed models of some components might be needed during the early design phase.

Thus, a trade-off between modeling depth and computational performance has to be found. This may be achieved by selectively simplifying parts of the model that are prone to generating stiff sub-systems or a large number of state events.

Within this framework, the present paper introduces the Modelica simulation model of TraXon, the new modular transmission for heavy commercial vehicles by ZF. The model can be adapted to various needs by replacing components according to the required modeling depth and/or dynamical behavior.

After a brief overview of the ZF in-house Modelica libraries and the architecture of the TraXon model, some approaches and tools are described for evaluating and optimizing models with respect to real-time issues.

Keywords: ZF Modelica libraries; model simplification; performance analysis; hardware-in-the-loop;

Vectorized single-track model in Modelica for articulated vehicles with arbitrary number of units and axles

Peter Sundström[1], Bengt Jacobson[2], and Leo Laine[3]

[1]Modelon AB
[2]Chalmers University of Technology
[3]Volvo Group Trucks Technology

A single-track model for articulated vehicles has been implemented in Modelica. Model equations are vectorized which allows the model to represent articulated vehicles with arbitrary number of units and axles without rewriting or adding new equations. Non-linear kinematic constraints are defined for coupling points between units making the model valid for large articulation angles.

Four use cases for the model are presented: Inverse dynamics for feedforward control, frequency responses when varying parameters, steady-state evaluations and dynamic simulation. For these use cases, four parametrizations of the model are used corresponding to a tractor with a semitrailer, a truck with a dolly and a semitrailer, an A-double (tractor+semitrailer+dolly+semitrailer) and an airport baggage carrier with five trailers. Figure 1 shows results for three of the use cases.

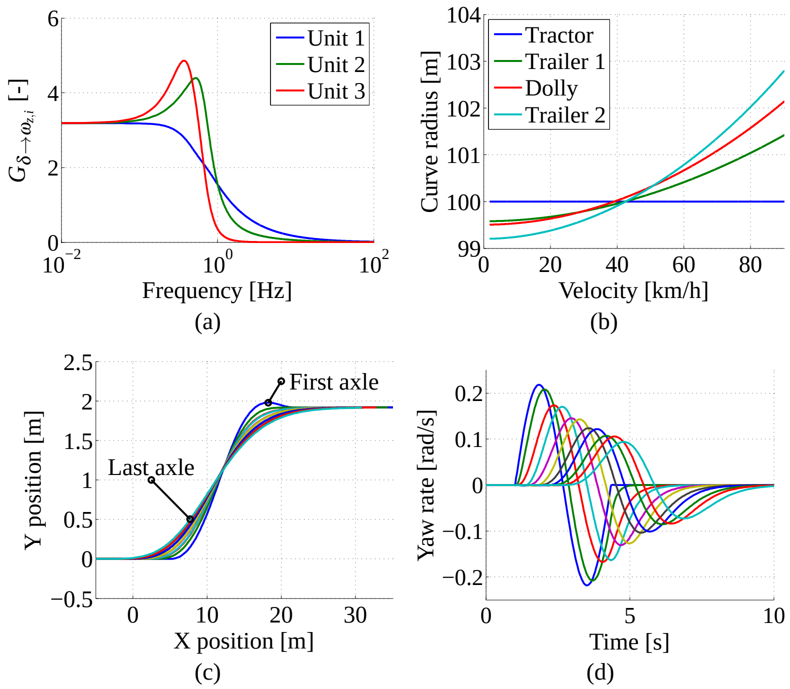


Figure 1: (a) Yaw rate frequency response of a three unit truck combination (b) Trailer steady-state off-tracking of a four unit truck combination (c) Axle positions of a five trailer vehicle in a lane change maneuver (d) Yaw rates of the 5 trailer vehicle in lane change maneuver

Multibody Model of a Motorbike with a Flexible Swingarm

Gianni Ferretti[1] Bruno Scaglioni[2] Andrea Rossi[2]
[1]Politecnico Di Milano, Dipartimento di Elettronica, Informazione
e Bioingegneria DEIB
Via Ponzio 34/5, 20133 Milano, Italy
[2]MUSP Lab, Via Tirotti 9, Le Mose, 29122 Piacenza, Italy

Considering the specific case of the multibody modelling of a racing motorbike, where the rigid model of the rear swingarm has been replaced with a flexible one, a general approach to flexible multibody systems modelling in Modelica is presented in this paper. In particular, the steps required to generate the model of a flexible body starting from a FEM analysis, performed with commercial packages, are detailed. Simulations results are shown with reference to a sudden braking and to a series of impacts with curbs. In this last case, an unstable behaviour occurred when considering the flexible component, which is currently under investigation.

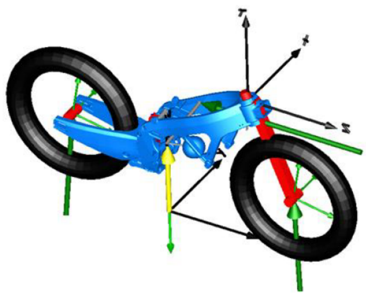


Figure 1: Model of the motorbike.

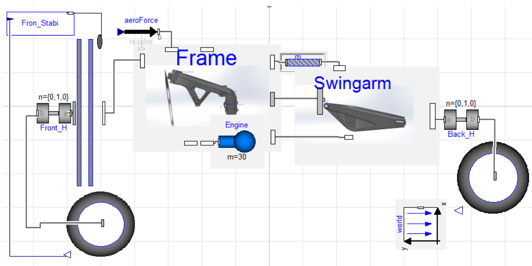


Figure 2: Scheme of the Modelica model of the motorbike.

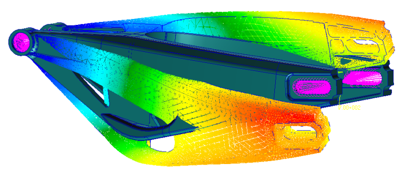


Figure 3: First torsional eigenmode.

Modelling and parameter identification of a semi-active vehicle damper

Michael Fleps-Dezasse   Jakub Tobolar   Johannes Pitzer

German Aerospace Center (DLR)
Institute of System Dynamics and Control
82234 Wessling
Michael.Fleps-Dezasse@dlr.de   Jakub.Tobolar@dlr.de
Johannes.Pitzer@dlr.de

In this paper two semi-physical models of the semi-active dampers of the DLR robotic electric vehicle ROboMObil (ROMO) are described and their implementation in Modelica is presented. Besides the damper characteristics and hysteresis, the models additionally consider the gas force and cover the differences of the damper characteristics for compression and rebound. A procedure to identify the damper model parameters was implemented using the DLR Optimization library. The measurement data used for parameter identification was recorded during experiments on a damper test bench. The simulation results of the damper models are compared to the experiment data of the semi-active damper and the suitability of the damper models with respect to accuracy and real-time simulation is discussed.

Figure 1 Semi-active dual tube damper with external continuously variable valve

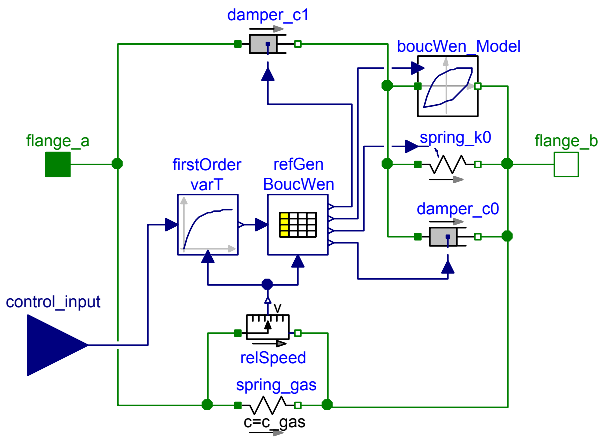


Figure 2 Generalized extended Bouc-Wen damper model

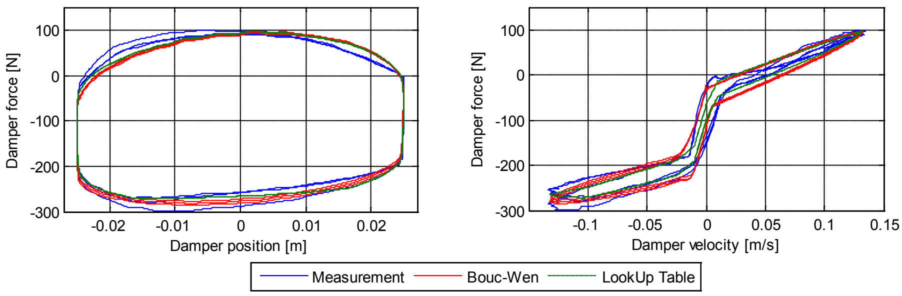


Figure 3 Comparison of damper models to damper measurements at 10% control input (left: Force over position; right: force over velocity)

The Modelica *HouseModels* Library: Presentation and Evaluation of a Room Model with the ASHRAE Standard 140

Ana Constantin Rita Streblow Dirk Müller
RWTH Aachen University, Institute for Energy Efficient Buildings and Indoor Climate
Mathieustr. 10, 52074 Aachen
{aconstantin, rstreblow, dmueller}@eonerc.rwth-aachen.de

In this paper we present the *HouseModels* library, which will be made available free of charge in summer 2014, as part of our contribution to the IEA Annex 60. The library contains complete standard house models for one and multi-family dwellings, as well as a model for a single apartment. The models can be easily parameterized for different thermal masses and energy saving ordinances. This variations are useful when testing energy concepts and control strategies, as a robust system has to be able to adapt to different types of buildings.

Our motivation for creating this library is to bridge the gap between developers and users of Modelica for dynamic building system simulations. The models are easy to understand and use. Extra effort has been made to enrich the parameter window and to make the icons dynamic in regard to the chosen parameters (figure 1). In order not to confuse first time users

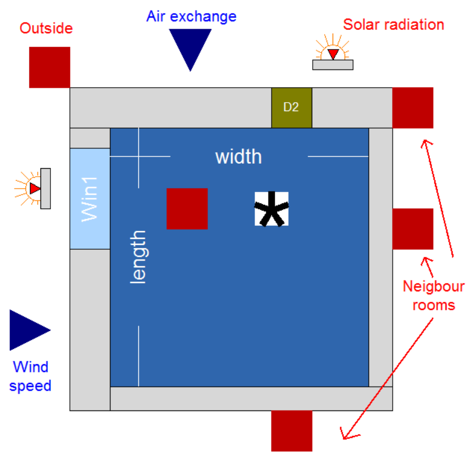


Figure 1: Icon for a room model

certain parameters have been set as protected and the parameterization of a room model can be done by specifying only a handful of parameters. This makes the library suitable for teaching purposes as well.

The set of room types developed for the one family dwelling can, if necessary, be parameterized differently than the standard model or extended in order to build up specific house models.

As we try to keep the models as simple as possible and as detailed as needed in order to have good CPU-times for the simulations, a validation of the room models is currently on the way. In this paper we present first results obtained with case 600 of the ASHARE Standard 140. For all the required outputs our room model produced results within the minimum and maximum specified ranges. We plan on further evaluating the models with the whole suite of tests and improving them if necessary.

Modelica Library for Building and Low-Voltage Electrical AC and DC Grid Modeling

Juan Van Roy[1,2,3] Robbe Salenbien[2,3] Johan Driesen[1,3]
[1] University of Leuven, Department of Electrical Engineering (ESAT)
Kasteelpark Arenberg 10, PB 2445, 3001 Leuven, Belgium
[2] Flemish Institution for Technological Research (VITO)
Unit Energy Technology, 2600 Mol, Belgium
[3] EnergyVille, 3600 Genk, Belgium

This paper presents a Modelica library for electrical grid systems for low-voltage distribution grids and in-building grids. The library is based on previous work, in which a library was presented to simulate fully balanced three-phase AC low-voltage distribution grids [1].

This library is extended to simulate three-phase unbalanced AC low-voltage distribution grids as part of the IDEAS library [2]. A second extension is the ability to simulate DC grids. The Modelica implementation also allows to simulate multiple types of grid (i.e. single-phase AC, three-phase AC and DC grids) on different scales (in-building or low-voltage distribution grids) in one simulation. For AC grid analyses, a quasi-stationary model is implemented, assuming a fixed frequency (e.g. 50 Hz). This allows to represent the waveforms by its amplitude and phase shift. Therefore, dynamic transient are not included.

Electrical grids may connect many different energy systems (loads and generation units), different grids and/or buildings within districts. The library allows to assess the grid impact of these systems on different grid types. Control or optimization strategies can use grid variables, such as voltages and power exchanges.

The electrical grid models are validated using a comparative validation method. The three-phase grid models from this Modelica library are compared with the power flow analysis tool in [3]. A simple case study is developed with a small three-phase unbalanced residential distribution grid, consisting of one feeder connected to the feeding transformer. The feeder consists of 20 residential loads connected to respectively one of the 20 nodes in this feeder. The comparative validation of this Modelica library shows that the difference in nodal voltages depends on the loads, the grid topology and end criterion. Also the mutual impedance of cables in three-phase systems is neglected. Nevertheless, the average voltage differences between both models are limited, for this case study in the order of 10^{-2} V for a voltage of 230 V.

References

[1] B. Verbruggen, J. Van Roy, R. De Coninck, R. Baetens, L. Helsen, and J. Driesen, "Object-Oriented Electrical Grid and Photovoltaic system modelling in Modelica," in *8th International Modelica Conference*, Dresden, Germany, Mar. 2011.

[2] R. Baetens, R. De Coninck, J. Van Roy, B. Verbruggen, J. Driesen, L. Helsen, and D. Saelens, "Assessing Electrical Bottlenecks at Feeder Level for Residential Net Zero-Energy Buildings by Integrated System Simulation," *Applied Energy*, vol. 96, pp. 74–83, Aug. 2012.

[3] J. Tant, F. Geth, D. Six, and J. Driesen, "Multiobjective Battery Storage to Improve PV Integration in Residential Distribution Grids," *IEEE Transaction on Sustainable Energy*, vol. 4, no. 1, pp. 182–191, Jan. 2013.

Tool coupling for the design and operation of building energy and control systems based on the Functional Mock-up Interface standard

Thierry Stephane Nouidui, Michael Wetter
Lawrence Berkeley National Laboratory
One Cyclotron Road, Berkeley, CA
TSNouidui@lbl.gov

This paper describes software tools developed at the Lawrence Berkeley National Laboratory (LBNL) that can be coupled through the Functional Mock-up Interface standard in support of the design and operation of building energy and control systems. These tools have been developed to address the gaps and limitations encountered in legacy simulation tools. These tools were originally designed for the analysis of individual domains of buildings, and have been difficult to integrate with other tools for runtime data exchange. The coupling has been realized by use of the Functional Mock-up Interface for co-simulation, which standardizes an application programming interface for simulator interoperability that has been adopted in a variety of industrial domains.

As a variety of coupling scenarios are possible, this paper provides users with guidance on what coupling may be best suited for their application. Furthermore, the paper illustrates how tools can be integrated into a building management system to support the operation of buildings. These tools may be a design model that is used for real-time performance monitoring, a fault detection and diagnostics algorithm, or a control sequence, each of which may be exported as a Functional Mock-up Unit and made available in a building management system as an input/output block. We anticipate that this capability can contribute to bridging the observed performance gap between design and operational energy use of buildings.

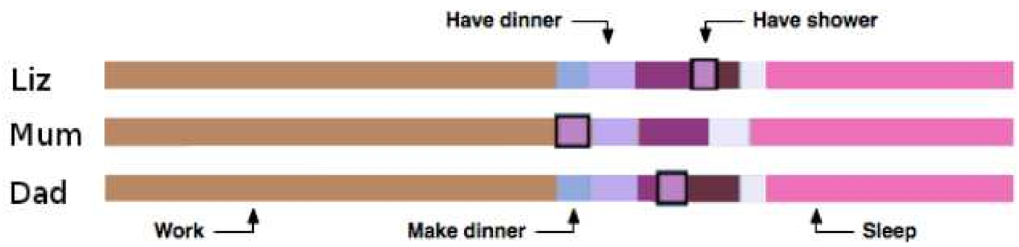
Coupling occupant behaviour with a building energy model - A FMI application

Gilles Plessis[1] Édouard Amouroux[2] Yvon Haradji[1]
[1]EnerBaT – EDF R&D, Moret sur Loing, FRANCE
[2]LIP6 – Université de Paris 6, Paris, FRANCE
gilles.plessis@edf.fr edamouroux@gmail.com yvon.haradji@edf.fr

Occupant behaviour is commonly described in dynamic building simulation tools using standardized occupancy profiles. Various studies suggest that occupant behaviour should be taken into account in a more accurate way, as it can have a dramatic impact on energy consumption especially in the context of low and positive energy buildings.

This paper illustrates the use of *Functional Mock-up Interface* (FMI) standard to couple an occupant behaviour simulator and a building model. Due to their intrinsic nature, occupant behaviour on the one hand, and a building and its energy systems on the other hand, are usually represented by different modelling paradigms. The occupant behaviour is here described by Agent-Based Modelling (ABM) whereas the building is described by a set of hybrid and differential algebraic equations, typical of dynamic thermal modelling. Such different complex systems cannot be efficiently simulated in a single tool. Therefore, one solution is the tool coupling approach.

The FMI standard for co-simulation was used to couple the SMACH occupant behaviour simulator and a building energy model built with the BuildSysPro Modelica library. Variables of interest are passed from one model to another at fixed synchronization time steps. The interactions that are considered between occupant behaviour on one side and the building and its energy systems on the other side are thermal comfort, occupants' control on HVAC systems and internal heat loads supplied to the building's ambient air temperature.



The first results of the co-simulation are presented. Activity diagrams, such as the one presented above, illustrate the variability of the occupants' actions in one given household: some activities are distinct and others can be synchronous, for instance "having dinner".

From a computing time perspective, the coupling is quite heavy. For a one month simulation and a time step of 1 minute, the occupant behaviour simulation takes 3 minutes with SMACH and the building energy simulation takes 10 seconds with Dymola. When coupling is applied, the same simulation takes 13 minutes.

Future work will address computer efficiency by considering variable size of communication steps and the modelling modularity of the building and its systems by composing FMUs.

Keywords: Building simulation; behavioural modelling; Specific use of electricity; thermal comfort; Modelica; FMI; co-simulation

Phenomenological Li ion battery modelling in Dymola

Kotub Uddin* and Alessandro Picarelli†

* *WMG, The University of Warwick,*
International Digital Laboratory, Coventry, CV4 7AL, United Kingdom

†*Claytex Services Ltd.,*
Edmund House, Rugby Road, Leamington Spa, CV32 6EL, United Kingdom

*k.uddin@warwick.ac.uk, †alessandro.picarelli@claytex.com

A key enabler (or constraint) of the electrified power train is the need to store energy in a form that can be easily and robustly converted into electricity. Batteries have emerged as a preferred choice in alternative energy storage but the technology still comes with significant compromises for the customer. Many of the challenges and opportunities presented by battery technology can be traced to the li-ion cell at the heart of the battery.

The structure of a modular, acausal and reconfigurable electro-thermal battery model is described. In this paper, we extend the INEEL FreedomCar program model [1] to include temperature dependence, voltage hysteresis, self-discharge and diffusion limitation. The dynamic model structure adopted for the battery cell is based on an equivalent circuit whose parameters are generated using real cycling data through an optimisation routine written in the Modelica language. A linearised one-dimensional thermal mathematical model with lumped parameters is used to simulate temperature profiles for the cell. The cell and scaled-up pack model is parameterised for a number of commercially available cells ranging a number of cell formats, sizes and chemistries. These Dymola models are validated using highly transient and aggressive real-world as well as synthetic drive cycles.
The same battery models can be tested on virtual rigs, integrated into vehicle powertrains thanks to the multi-domain properties of the Modelica language on which Dymola is based.

Keywords: Lithium ion, battery, HEV, EV, PHEV, Acausal, Dymola, Modelica

References

[1] I.N.E.E. Laboratory, Battery Test Manual for Plug-In Hybrid Electric Vehicles, in, Assistant Secretary for Energy Efficiency and Renewable Energy (EE), Idaho Operations Office: Idaho Falls, ID, USA, 2010., 2010.

A Modelica Based Lithium Ion Battery Model

Johannes Gerl [a] Leonard Janczyk [a] Imke Krüger [a] Nils Modrow [a]
[a] Modelon GmbH
Agnes-Pockels-Bogen 1
D-80992 München

johannes.gerl@modelon.com leonard.janczyk@modelon.com
imke.krueger@modelon.com nils.modrow@modelon.com

Keywords: *battery model; lithium-ion; behavioral modeling; electrical vehicle*

In Battery Electric Vehicles (BEV) and Hybrid Electric Vehicles (HEV) the majority of car producers focus in lithium ion based battery concepts due to their high performance density in connection with reasonably high lifetime and acceptable thermal behavior. As these vehicles become more accepted on the market, the production numbers are supposed to increase with some positive pricing effect. It is likely that this will also make lithium ion batteries attractive for use in homes and other decentralized energy systems – especially in connection with renewable energy.

Practically all lithium ion based batteries show more or less troublesome aging behavior which reduces the lifetime to inacceptable levels, if no particular provisions are taken to avoid or reduce it. Aging appears as calendric and as cyclic effect according to the number of charging and re-charging events.

Aging effects are severely influenced by the thermal load on the battery. Therefore high performance battery systems need to be kept within a certain temperature range by cooling and sometimes heating.

For whatever application, in current battery systems single cells of a certain type are arranged in stacks, modules or packages through serial and parallel alignment of the cell. Cells can have cylindrical, prismatic or so-called coffee bag shape. Apart from the electrical interconnection, the cells are integrated in some thermal design concept to cool them and reduce aging. It should be noticed that car as well as energy system manufacturers design battery systems according to the needs of the general concept of their product. I.e. the design of the battery is not based on a unified single-type approach, but many different concepts are required to cover the large range of system requirements.

Therefore, the battery model presented in this paper uses the cell as a base unit to be parameterized with fairly simple data sheet and empiric input. With the help of pre-defined templates the user can easily set-up a battery model as an electrical and thermal system consisting of a single cell. Aging information is provided by an integrated aging system or user-defined approach.

While lithium ion battery cells are usually described by RC circuit elements, the electrochemical effects in lead-acid batteries are approximated in a separate model to take account of the specialties of this battery type.

In the following, a comprehensive Modelica model is introduced for the simulative description of the physical behavior of lithium ion battery cells packs for relevant aspects and use cases. It is part of the Modelon Battery Library, a commercial Modelica library to model battery cells and packs of various types, shape and grouping.

Thermal behavior, electrical behavior and the impact of the degradation due to aging are considered as they influence each other.

The model parameters to calculate the electrical behavior are to be derived from measurements; an optimization algorithm to obtain them is integrated in the package using the Optimization Library. Functions to validate the model against these measurements are included as well.

As an application example the simulation of an energetic energy storage system in the model of a battery electrical vehicle is shown.

Behavioral Modeling of Power Semiconductors in Modelica

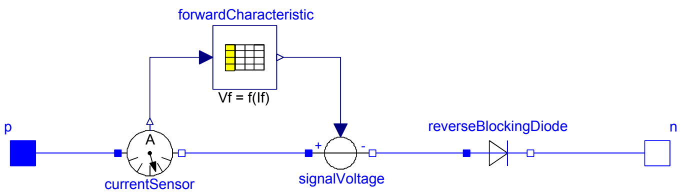
Patrick Denz
Vorarlberg Univ. of Appl. Sc.
Dornbirn/Austria
Patrick.Denz@omicron.at

Thomas Schmitt
Modelon GmbH
Munich/Germany
Thomas.Schmitt@modelon.com

Markus Andres
Modelon GmbH
Munich/Germany
Markus.Andres@modelon.com

The paper introduces behavioral (macro) models of power semiconductors, i.e. diodes, MOSFETs and IGBTs, being part of a library for simulating power electronics utilized, e.g. in electrified powertrains of either hybrid electric vehicles (HEV) or purely battery electric vehicles (BEV). The models consider static, dynamic (switching mode) and thermal effects and in most cases can be fully parameterized solely on the basis of characteristic curves and parameters specified in datasheets. The main purpose of behavioral models is an accurate representation of the semiconductor signals to, e.g. calculate the overall losses. To this end the component's behavior is described via characteristic curves and parameters provided in datasheets.

In the simplest case, the static model of a diode can be described as depicted in the figure below. The current through the diode is measured using a current sensor and serves



as input signal to a table which stores the forward characteristic $V_f = f(I_f)$ specified in the datasheet. The corresponding forward voltage drop is fed into a signal-controlled voltage source. The additional ideal reverse blocking diode ensures that the current solely flows in forward direction. Thus, on the one hand behavioral models of, e.g. diodes and MOSFETs can be parameterized solely on the basis of datasheets and on the other hand, the models behave as specified by the manufacturer under nominal conditions. In case of IGBTs due to their internal semiconductor structure, the occurring tail current has to be measured in advance. Moreover, in trench/field-stop IGBTs due to the additional field-stop layer added to the semiconductor structure the model developed in [1] is not valid anymore and has to be modified.

The MOSFET models are verified in simulations with various test circuits and are validated with measurement data provided by a company developing electric drive systems.

Since behavioral models provide detailed switching slopes, the simulation performance is totally unacceptable if such models are used to simulate, e.g. the driving range of an electric vehicle. The arising numerical problems are discussed and possible solutions are provided on how to modify the models in order to use them in e.g. system simulation.

References

[1] Wintrich, Arendt: Verhaltensmodellierung von Leistungshalbleitern für den rechnergestützten Entwurf leistungselektronischer Schaltungen, Technische Universität Chemnitz, 1996.

Verification and Design Exploration through Meta Tool Integration with OpenModelica

Zsolt Lattmann[2], Adrian Pop[1], Johan de Kleer[3], Peter Fritzson[1], Bill Janssen[3], Sandeep Neema[2], Ted Bapty[2], Xenofon Koutsoukos[2], Matthew Klenk[3], Daniel Bobrow[3], Bhaskar Saha[3], Tolga Kurtoglu[3]

[1]Department of Computer and Information Science
Linköping University, SE-581 83 Linköping, Sweden
[2]Vanderbilt University, Nashville, TN, USA
[3]Palo Alto Research Center, 3333 Coyote Hill Road, Palo Alto, CA 94304 USA

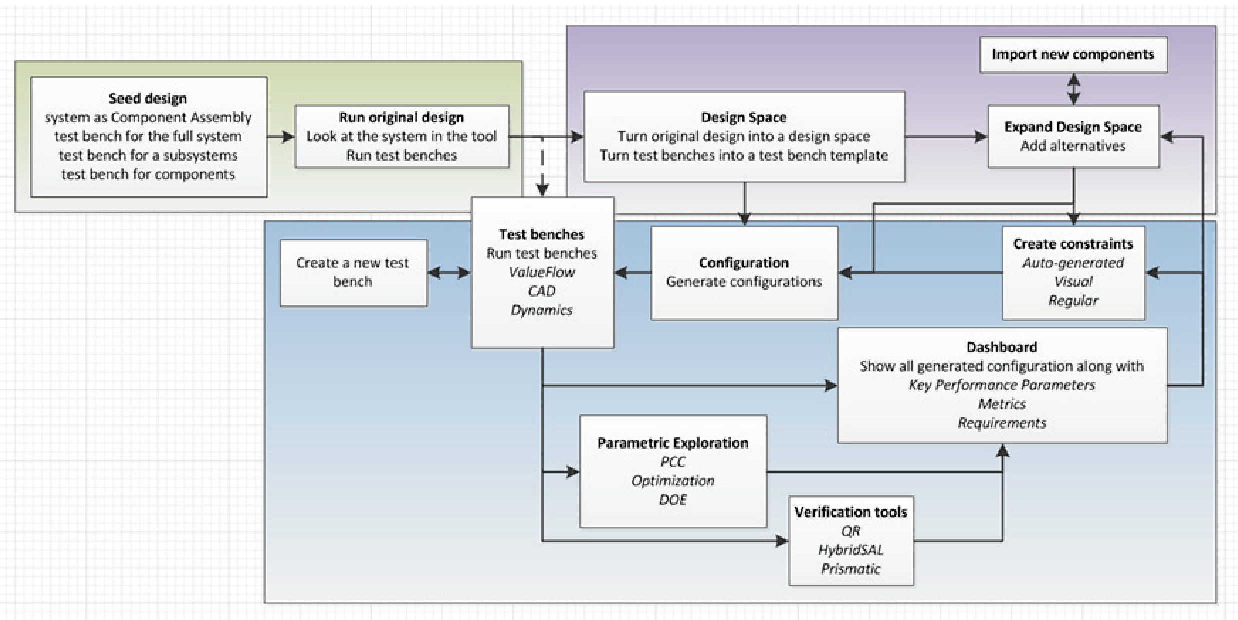
lattmann@isis.vanderbilt.edu, {adrian.pop, peter.fritzson}@liu.se,
dekleer@parc.com

Modelica models are typically used for simulation to investigate properties of a possible system designs. This is often done manually or combined with optimization to select the best design parameters.

It is desirable to have systematic and partly auto-mated support for exploration of the design space of possible designs and verifying their properties vs. requirements. The META design tool chain is being developed to support this goal. It provides an integration framework for components, designs, design spaces, requirements, and test benches, as well as verification of requirements for the generated design models during design exploration

This paper gives an overview of the META tools and their integration with OpenModelica. The integrated environment currently has four main uses of OpenModelica: importing Modelica models into the META tool model structure, performing simulations within test benches, analyzing Modelica models and automatically adding fault modes, and extracting equations (DAEs) for formal verification tools, e.g. the QRM using qualitative reasoning.

A prototype of the integrated tool framework is in operation, being able to generate and simulate thou-sands of designs in an automated manner.



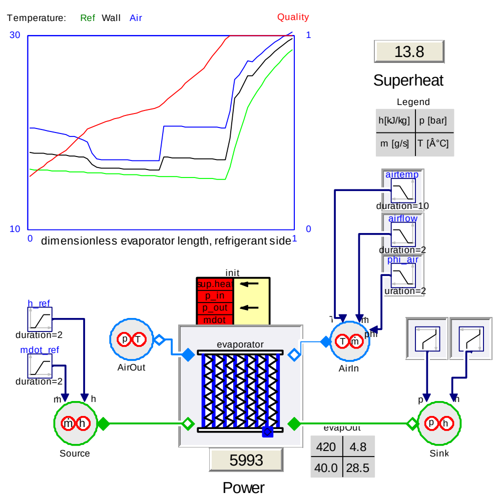
Parallel Model Execution on Many Cores

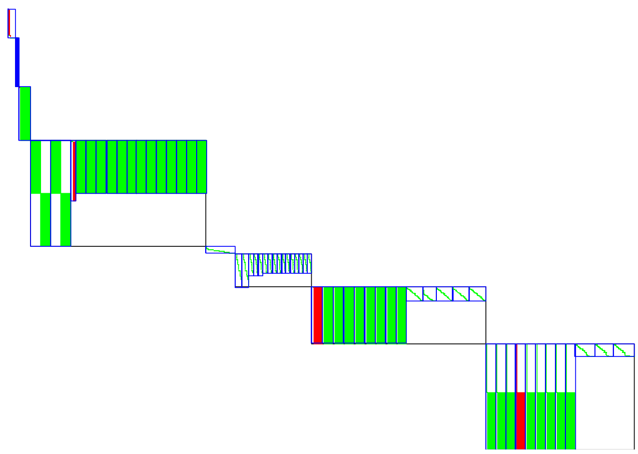
Hilding Elmqvist Sven Erik Mattsson Hans Olsson
Dassault Systèmes
Ideon Science Park, SE-223 70 Lund, Sweden
Hilding.Elmqvist@3DS.com SvenErik.Mattsson@3DS.com
Hans.Olsson@3DS.com

Modelica gives the possibility to compose more and more detailed models since model components can be reused. This means that simulation needs to be faster. One possibility is then to use multi-core technology. Recent advances with more than 1000 cores show the potential.

The problem is then how to utilize this enormous processing power in a user friendly way. Partitioning needs to be made automatically. Modelica gives good possibility to automatically partition the model equation execution into separate threads since it is a declarative language based on equations.

This paper describes a method to automatically parallelize model equations implemented in Dymola. A speed-up of 3.4 times has been achieved using 4 cores/8 threads.





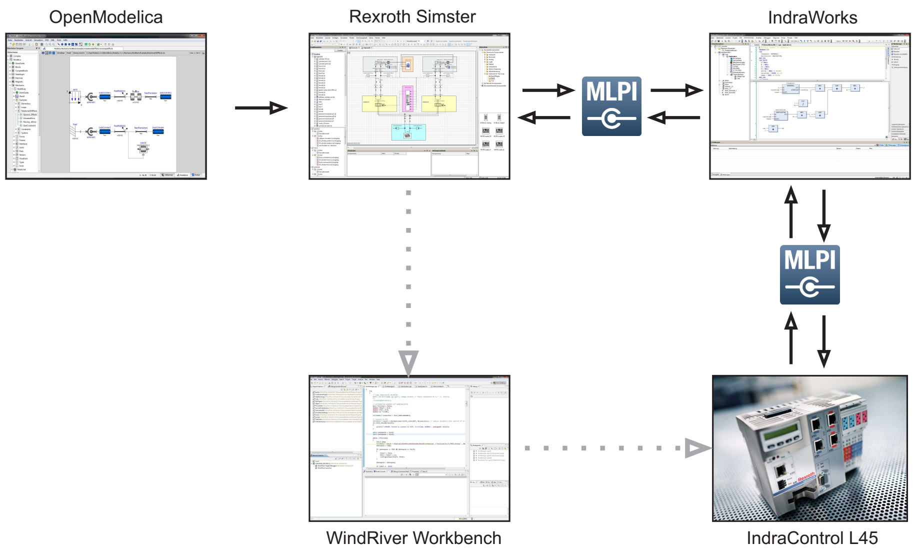
Parallel Schedule for Evaporator execution on max 16 cores

A toolchain for Rapid Control Prototyping using Rexroth controllers and open source software

Nils Menager   Niklas Worschech   Lars Mikelsons
Bosch Rexroth AG
Rexrothstr. 3, 97816 Lohr a. Main

Taking a look at project costs from a financial point of view, the commissioning times of new industrial systems become more and more important, as they significantly drive the costs. Hence, the reduction of commissioning times is part of current research. Besides the use of simulation and the coupling between hardware and software (Hardware-In-The-Loop-Simulations), *Rapid Control Prototyping* offers a huge potential to reduce commissioning times. There are already possibilities to use virtually designed control algorithms on real-time operating systems using Hardware-In-The-Loop-Setups. Two possibilities are using a dSPACE system or a xPC system in combination with *Matlab/Simulink*. Though, using such a toolchain leads to three serious drawbacks.

First, these commercial systems are very expensive. Second, even in that case the control algorithm has to be re-implemented on the real control hardware after testing on the real-time system. Furthermore, it has to be taken into account that the usage of such commercial real-time systems leads to dependencies to external software (e.g. *Matlab/Simulink*).



The aim of this work is to set up a toolchain for Rapid Control Prototyping with a Rexroth controller (IndraControl L45) using open source software and Modelica for the modeling part, i.e. a toolchain, which is completely independent from external software and hardware. To be more precise, this toolchain enables the engineer to transfer virtual controller models modeled in Modelica directly to standard Rexroth controllers. To validate the functionality, in this contribution, the controller is used in a Hardware-In-The-Loop setup. The validation of the controller in combination with a real system is part of current work.

Modular Multi-Rate and Multi-Method Real-time Simulation

Bernhard Thiele† Bernhard.Thiele@dlr.de
Martin Otter† Martin.Otter@dlr.de
Sven Erik Mattsson‡ SvenErik.Mattsson@3ds.com

† German Aerospace Center (DLR), Institute for System Dynamics and Control, 82234 Wessling, Germany
‡ Dassault Systèmes AB, Ideon Science Park, SE-223 70 Lund, Sweden

The demand to ever increase realism and scope of models routinely exceeds the currently available computing power and thus requires thoughts on improving simulation efficiency. This is especially true for real-time simulations, where fixed timing constraints do not allow to just “wait a bit longer”.

This paper presents a new approach in Modelica that allows a modeler to separate a model into different partitions for which individual solvers can be assigned. In effect, this allows to use multi-rate and multi-method time integration schemes that can contribute to improve the efficiency of a (real-time) simulation.

The first part of the paper discusses basic consideration relating to modular (real-)time integration. Afterwards, the implementation of a convenient Modelica library for the partitioning of physical models is briefly described. Finally, the presented library is used to partition a detailed six degree of freedom robot model for modular simulation. The simulation performance of that partitioned model is compared to the simulation performance achieved by using “conventional” global solvers.

Figure 1 shows the robot benchmark example which is based on the `RobotR3` example from the Modelica Standard Library. It is partitioned into three parts: a) a discrete-time (digital) *control partition*, b) a *drive partition*, solved by an implicit Euler method, and c) a *multi-body partition*, solved by an explicit midpoint method. The coupling between b) and c) is achieved by using modeling elements from the presented library.

Keywords: multi-rate / multi-method time integration; simulation; clocked discretized continuous-time partitions.

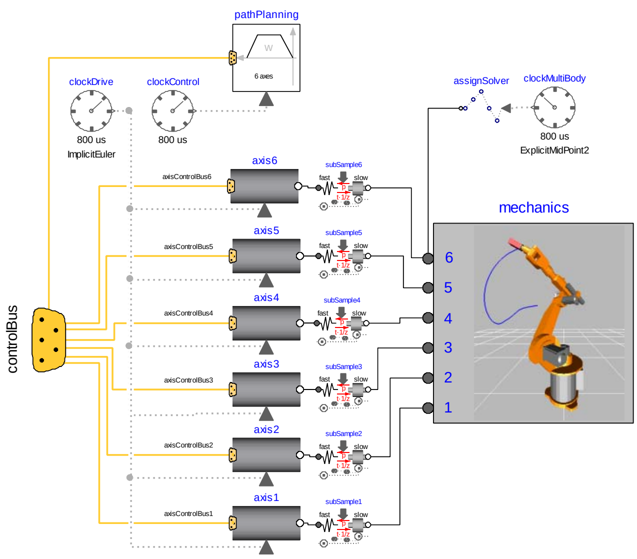


Figure 1: 6-DOF robot example partitioned into three parts. The respective solvers are assigned using the components `clockControl`, `clockDrive`, and `clockMultiBody`.

Significant Reduction of Validation Efforts for Dynamic Light Functions with FMI for Multi-Domain Integration and Test Platforms

Dr. Stefan-Alexander Schneider   Johannes Frimberger
BMW AG, 80788 Munich, Germany
stefan-alexander.schneider@bmw.de, Johannes.JF.Frimberger@bmw.de

Michael Folie
IPG Automotive GmbH, Agnes-Pockels-Bogen 1, 80992 Munich, Germany
michael.folie@ipg.de

Modern advanced driver assistance systems (ADAS) provide a significant increase in comfort and safety. In many cases, a single vehicle, today, contains more than one assistance system, while the trend to use ADAS continues to grow. At the same time, the number of systems that perform control interventions with safety-relevant functions increases as well. From an overall perspective, it must be assured that the driver assistance systems – individually as well as in the way they interact – function flawlessly in any driving situation and with any driver at the wheel anywhere in the world. This results in an increased development and testing effort for modern ADAS in general. In-development testing based on virtual test driving offers an approach to a solution that allows the validation effort to be significantly reduced while meeting the requirements for safety-relevant functions in the vehicle associated with ISO 26262. This is particularly evident when developing and testing new light functions, which in real-world road tests can often be performed only in conditions of darkness. Dynamic Light Functions are defined as the situation-dependent headlight adjustment consisting of cornering light, headlight leveling and (glare-free) headlight assistance. This paper presents this new methodology.

References

[1] Schick, B.: Mission V-Process enhancement by integrated vehicle performance evaluation within an entire X-in-the-Loop process, Keynote SIAT ARAI 2013

[2] FMI Development Group: FMI – The Functional Mock-up Interface. [https://www.fmi-standard.org/]: [January 21, 2014]

[3] AUTOSAR Development Partnership: AUTOSAR, [http://www.autosar.org/]: [January 21, 2014]

[4] Schneider, S.-A., B. Schick and H. Palm: Virtualization, Integration and Simulation in the Context of Vehicle Systems Engineering, embedded world 2012, Nuremberg 2012

[5] BMW UK: BMW Intelligent Headlight Technology: Long Version: [http://www.youtube.com/watch?v=-dvPZ3H1Vm4]: [January 21, 2014]

[6] Palm, H., Holzmann, J., Schneider, S.-A., Kögeler, H.M. and Pfister, F.: The Future of Car Design: Optimisation with Systems Engineering, ATZ Elektronik 06/2013

Hardware-in-the-Loop (HIL) Simulation with Modelica – A Design Tool for Thermal Management Systems

S. Baltzer T. Lichius J. Gissing P. Jeck L. Eckstein
Institute for Automotive Engineering (ika) – RWTH Aachen University
Steinbachstraße 7, 52074 Aachen
baltzer@ika.rwth-aachen.de

Jörg Küfen
Forschungsgesellschaft Kraftfahrwesen mbH
Steinbachstraße 7, 52074 Aachen
kuefen@fka.de

Due to the higher complexity of electrified vehicles the requirements for vehicle components and vehicle design augment and new development tools are desirable. The following paper describes the design of a hardware-in-the-loop (HIL) test bench along with its structure using Modelica and a Remote Process Communication library. The aim is to support the development of components and operational strategies under realistic boundary conditions. The test bench is planned and built up within the scope of the public founded project qOpt at the Institute for Automotive Engineering (RWTH Aachen University) in cooperation with the Forschungsgesellschaft Kraftfahrwesen mbH Aachen and the Institute of Automatic Control (RWTH Aachen University). An overview of the participated applications currently in use at the HIL is shown in Fig. 1.

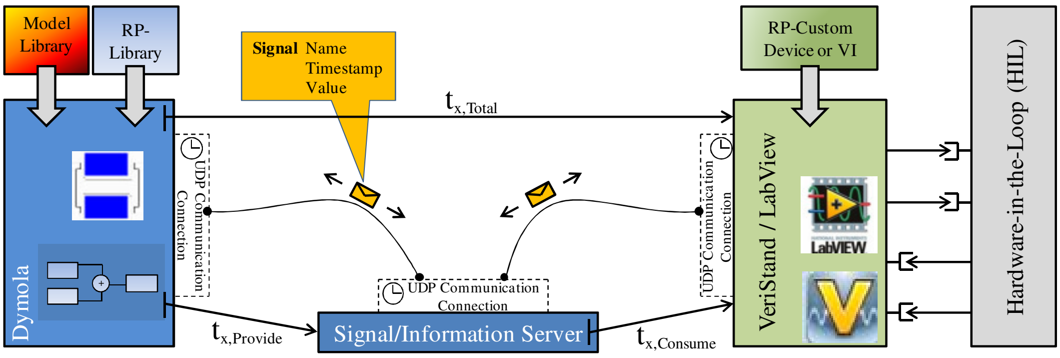


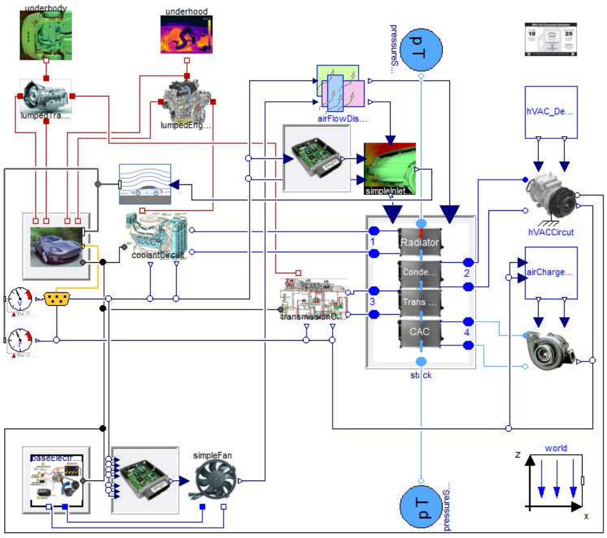
Fig. 1: Software structure of the HIL system. Multiple applications are joined for operation.

An application example is given in which a thermal storage unit is integrated into a Plug-In-Hybrid Electric Vehicle, to enhance the combined heat and power (CHP) usage of the internal combustion engine (ICE). In order to measure the usable amount of waste heat of the ICE the vehicle is placed on a dynamic chassis dynamometer and the cooling circuit of the ICE is connected to the HIL. All heat sinks, like the passenger cabin as well as the thermal storage are simulated in Dymola and the respective physical values are transferred to the thermo-hydraulic test bench. The results show that a reasonable amount of waste heat could be recovered in a thermal storage. Therefore, the electric driving range can be enhanced by providing the heat demand of the passenger cabin in form of the storage unit.

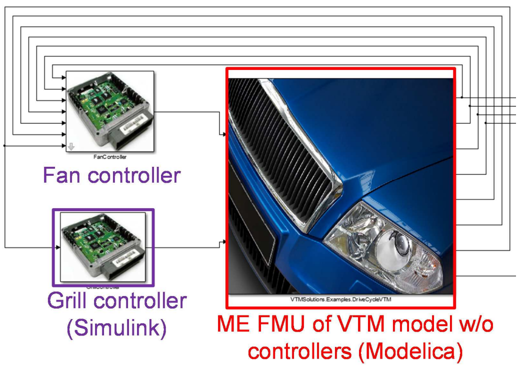
Integrated Vehicle Thermal Management in Modelica: Overview and Applications

John Batteh[1] Jesse Gohl[1] Sureshkumar Chandrasekar[2]
[1] Modelon, Inc. Ann Arbor, MI, USA
[2] Modelon, Inc. Hartford, CT, USA
{john.batteh, jesse.gohl, chandrasekar.sureshkumar}@modelon.com

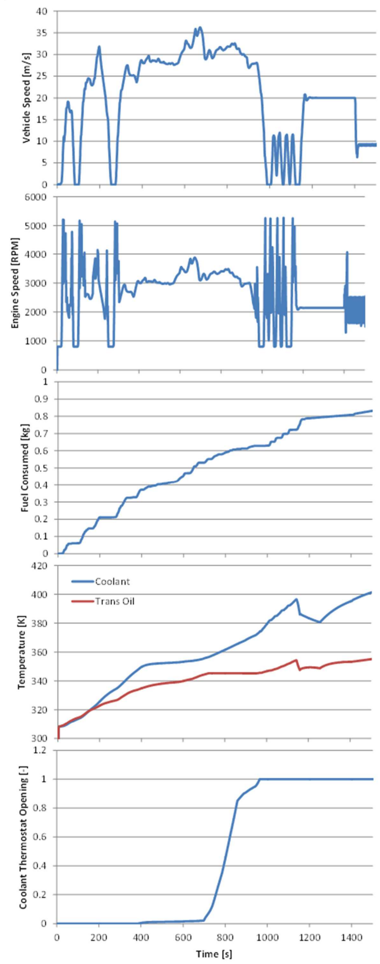
This paper highlights the use of a coordinated suite of Modelica libraries for vehicle thermal management (VTM) applications. The models are implemented using the Vehicle Dynamics Library, Liquid Cooling Library, and Heat Exchanger Library from Modelon. An integrated vehicle thermal management model is implemented, including the key physical and controls models. The model is used to demonstrate complex, multi-domain interactions between the physical and control systems over drive cycles for combined thermal and fuel efficiency studies. The model is also used to support controller development and optimization as an FMU integrated into Simulink. The flexibility of FMI-based workflows is also illustrated via batch and Monte Carlo simulations in Excel. A heat exchanger application coupling inputs from CFD illustrates the use of higher fidelity models from Heat Exchanger Library for calculation of performance degradation due to non-uniformity.



IntegratedVTM model



VTM model integrated with controls in Simulink as an FMU



Sample drive cycle results

Virtual Integration for hybrid powertrain development, using FMI and Modelica models

Lionel Belmon[1] Yujung Geng[1] Huaqiang He[2]

[1]: Global Crown Technology, Beijing, China
Lionel.Belmon@globalcrown.com.cn

2: Dongfeng Commercial Vehicles Technical center, Wuhan, China
hehuaqiang@dfcv.com.cn

Dongfeng Commercial Vehicles (DFCV) is developing powertrain controls for a hybrid light truck. To support this development, a virtual integration platform is being implemented, using Modelica models and Functional Mock-up Units (FMUs) for the engine/EMS, gearbox, MCU/e-motors, driveline, tyres and longitudinal dynamics. Simulink models and/or c-code of the Hybrid Control Unit (HCU) and Transmission Control Unit (TCU) are also integrated in the platform to achieve closed-loop simulation. The virtual integration allows reproducing accurately the overall vehicle behavior and is used for optimization of gearshifts, hybrid mode switches and hybrid drive strategies.

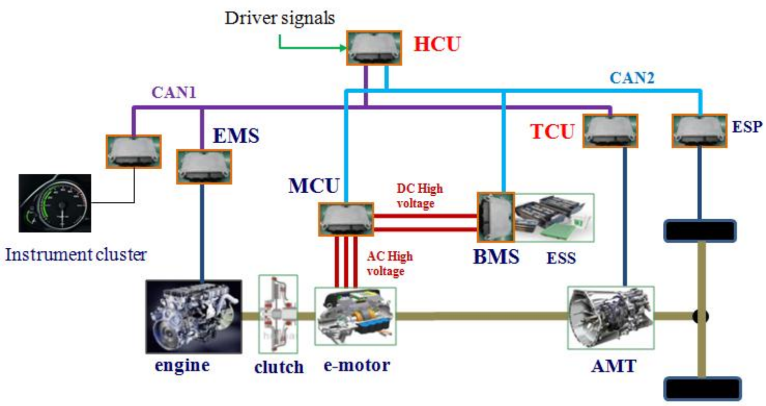


Figure 1 : schematic of the hybrid powertrain

References

[1] A.Abel, T.Blochwitz, A.Eichberger et al. Functional Mock-up Interface in Mechatronic gearshift simulation for commercial vehicles, 9th International Modelica Conference, 2012, Munich.

[2] A.Junghanns, R.Serway, T.Liebezeit, M.Bonin. Building virtual ECUs quickly and economically. ATZ Elektronik, 03/2012, Volume7.

[3] E.Chrisofakis, A.Junghanns. Simulation-based development of automotive control software with Modelica. Dresden: Modelica international conference, 2011.

[4] M.Tatar, Schaich, Breitinger. Automated test of the AMG speedshift DCT control software. Berlin: 9th CTI Innovative Automotive Transmissions Symposium, 2010.

General fault triggering architecture to trigger model faults in Modelica using a standardized blockset

Francicus L. J. van der Linden
franciscus.vanderlinden@dlr.de

German Aerospace Center (DLR), Institute of System Dynamics and Control,
82234 Weßling, Germany

Adding faults to Modelica models can be done in many different ways and is not standardized in any way. To support the standardization of Fault modeling in Modelica, a set of blocks is presented using valid Modelica code which are used are sources for Faults.

The proposed method support the injection of parameter faults (constant during simulation time) as well as variable faults. Special care has been taken that in absence of a block to control the faults, the default values (no fault) are used.

A trade of study between multiple fault injection options is made. Based on the results of this study, a broadcasting mechanism based on the inner-outer system is selected to control the faults. This minimizes user interaction during model building and does not need any non-physical connections. Also upgrading system components with extra faults does not need any model redesign. Using the abstract syntax tree (AST) functionality of the Dymola `ModelManagement` library, the model is analyzed and all faults in the model and its sub models are found. Using the automatically generated wrapper library, it is possible to conveniently control all faults in the model.

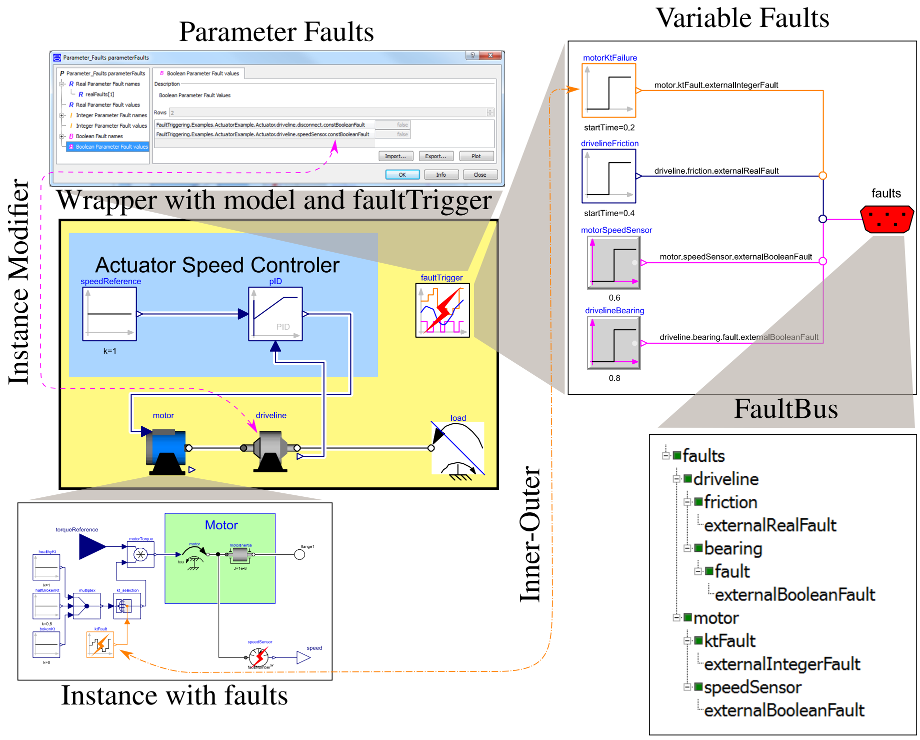


Figure 1: Automatically generated wrapper model (yellow) which contains the extended original model and the block `faultTrigger`. In this block all parameter and variable faults can be set. The parameter faults communicate directly with the model instances using instance modifiers (pink dash-dotted line), the variable faults using a bus system connected to a global inner/ outer system (orange dashed line).

Using Fault Augmented Modelica Models for Diagnostics

Raj Minhas Johan de Kleer Ion Matei Bhaskar Saha
Bill Janssen Daniel G. Bobrow Tolga Kurtoglu
Palo Alto Research Center
3333 Coyote Hill Road, Palo Alto, CA 94304 USA
contact email: dekleer@parc.com

We propose a model-based diagnosis framework in which Modelica models of faulted behavior are used in combination with a Bayesian approach. The fault augmented models are automatically generated through a process developed as part of our Fault Augmented Model Extension (FAME) work. Fault diagnosis using a Bayesian approach is based on computing a set of probability density functions, a process that is usually intractable for any reasonably complex system. We use Approximate Bayesian Computation (ABC) to bound the numerical and computational complexity. The basic idea is to use fault augmented Modelica models to create probability distributions of possible outcomes and then compare those distributions against actual observations to perform parameter estimation. The detection of faults is treated as a model selection problem and the inference of their severity levels is treated as parameter estimation. The diagnostic precision of this approach is evaluated on a Modelica vehicle drive line model.

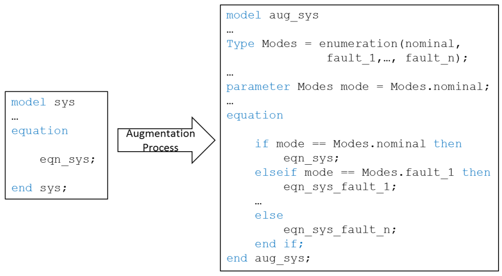


Figure 1: Alternative dynamics are enabled for each operating mode.

In addition to predicting behaviors through numerical simulations of Modelica models, we propose to use the same simulation models for diagnosis. Modelica's focus on simulation would seem to make it a poor choice for diagnosis. After all, diagnosis is the inverse of simulation. Simulation predicts the behavior of a system given a (correct) model. Diagnosis must infer how the model has changed (i.e., faulted) from observed behavior.

Most model-based diagnosis algorithms perform inference on declarative models. Although Modelica supports the writing of declarative models, too many Modelica models (including many in the MSL) contain imperative constructs making direct application of existing model-based diagnosis algorithms problematic. RODON is a Modelica inspired approach to modeling, but Modelica models first have to be re-written by hand in qualitative declarative form. We know of no system identification or FDI technique that applies for DAE models with boolean constraints (as Modelica models translate into). Our approach, on the other hand, applies on Modelica models directly no matter what types of constraints they contain.

From Modelica Models to Fault Diagnosis in Air Handling Units

Raymond Sterling[1], Peter Struss[2], Jesús Febres[1], Umbreen Sabir[2], Marcus M. Keane[1]
[1]Informatics Research Unit for Sustainable Engineering, Ryan Institute, NUI Galway, Ireland
[2]Computer Science Department, Technische Universität München
raymond.sterling@nuigalway.ie, struss@in.tum.de

Heating Ventilation and Air conditioning (HVAC) systems are known for being very inefficient for different reasons, one of the most common causes being the presence of undetected failures in one or more of its components. Undetected faults can remain for long periods due to different factors: compensations made by the control algorithms of other elements belonging to the same system; lack of proper maintenance, improper timing of flow of energy to/from the building, etc. Even when systems are known to suboptimal operation, the presence of faults may be very difficult to manually localize and identify, making it a costly task for human operators who only act when indoor environmental conditions are not met. This lack of timely intervention raises the need for developing automated fault detection and diagnosis methods and technologies that assist the building operator.

Different fault detection and diagnosis (FDD) methodologies have been developed for HVAC systems, mostly based on expert knowledge to help identifying the faulty condition and its source. However, a new trend in FDD is that of using models of the HVAC systems providing a base line for optimal operation, and supporting the detection of deviation from this optimum. Model-based methods, offer the advantage of an increased flexibility to adapt to different and innovative HVAC systems. Model-based diagnosis is based on an explicit representation of the knowledge about the components and the information about the plant structure, which determines how the components interact with each other. Unlike other approaches, the development of model-based diagnosis system can be fully automated rather than hand-tailored once a suitable component library and the plant structure is place.

This paper presents an end-to-end tool chain for a model-based diagnostic solution that uses a **qualitative model** for the part of the HVAC system corresponding to the **Air Handling Unit** (AHU). This solution is derived from a general first-principle Modelica model and exploits a general diagnosis algorithm that isolates and identifies faults that occur frequently and can cause significant loss of system performance in AHUs: passing heating- and cooling-coil valves, and stuck dampers. An application example using a heating coil model is presented where the diagnosis solution is able to correctly recognise a passing heating valve. Provisions are made for the extension to other components where the same methodology can be applied only requiring first principle models that accurately represent each component behaviour.

Simulation for verification and validation of functional safety

Lars Mikelsons Zhou Su
Bosch Rexroth AG
Rexrothstr. 3, 97816 Lohr am Main

Safety of machinery is the most critical issue in the design of mechatronic systems. The verification and validation procedure for functional safety of machinery is thoroughly discussed in ISO 13849-2. Following this procedure, the system behavior in case of a component failure has to be analyzed. Up to now this analysis bases on expert knowledge and real experiments. In this contribution a simulation based approach is presented. This approach has several advantages over the state-of-the-art. First, real experiments are more time consuming and costly than simulation. Moreover, according models can be used for further investigations like optimizing the sensor setup.
To enable failure simulation as a substitute of testing on real machinery for validation of functional safety, typical hydraulic failures are added to safety-related components of an in-house Modelica hydraulics library. This library is then used for the verification and validation of functional safety of a hydraulic test bench. Moreover, error propagation is considered.

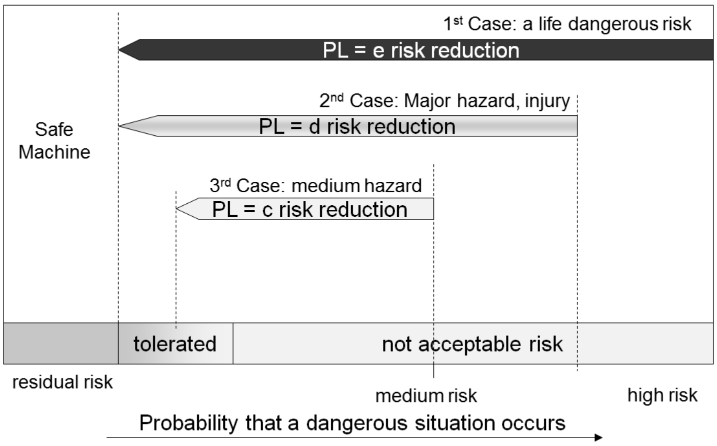


Figure 1: Principle of the risk reduction by the safety function

The Foundation of the DLR RailwayDynamics Library: the Wheel-Rail-Contact

Andreas Heckmann◇, Alexander Keck◇, Ingo Kaiser◇, Bernhard Kurzeck◇
◇German Aerospace Center (DLR), Institute of System Dynamics and Control
Oberpfaffenhofen, 82234 Wessling, Germany
andreas.heckmann@dlr.de

The formulation of the wheel-rail contact is a crucial issue in simulations considering the running dynamics of railway vehicles. Therefore a modeling environment that is dedicated to railway vehicle dynamics such as the new DLR RaiwayDynamics Library relies on an efficient representation of the kinematics and forces or torques, respectively, that appear at the wheel-rail interface. In order to give an impression on the geometry and the physics Fig. 1 presents a wheel-rail contact patch and the associated normal stress distribution.

A number of different formulations have been developed since the underlying rolling contact problem was firstly discussed in literature in 1876 [1]. The paper overviews these wheel-rail contact formulations in order to motivate the model, that has been chosen for implementation, namely a quasielastic ellipitical single point contact [2] with tangential force law according to *Polach* [3]. This model is then presented in detail.

Finally, the DLR RailwayDynamics Library is used to model and simulate the behavior of an experimental scaled M 1:5 running gear operating on the DLR roller rig, see Fig. 2. The simulations results are compared and validated with measurements.

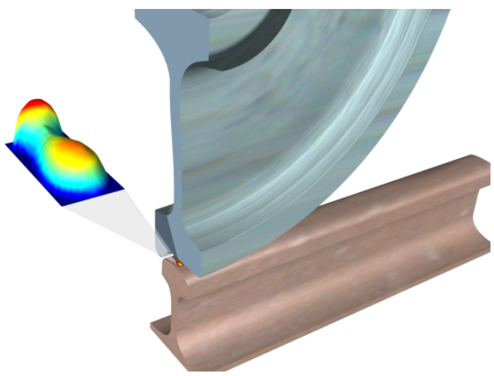


Figure 1: Examplary wheel-rail contact patch and the associated normal stress distribution

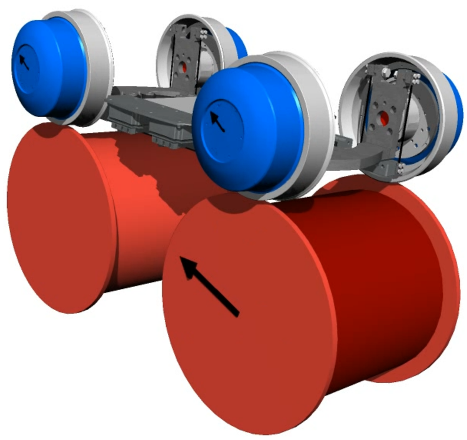


Figure 2: An animation of DLR's M 1:5 roller rig with experimental running gear

References

[1] K. Knothe. History of wheel/rail contact mechanics: from Redtenbacher to Kalker. *Vehicle System Dynamics*, 46(1-2):9–26, 2008.

[2] H. Netter, G. Schupp, W. Rulka, and K. Schroeder. New aspects of contact modelling and validation within multibody system simulation of railway vehicles. *Vehicle System Dynamics*, 29(S1):246–269, 1998.

[3] O. Polach. A fast wheel-rail forces calculation computer code. *Vehicle System Dynamics*, 33:728–739, 2000.

Human-Nature Interaction in World Modeling with Modelica

Rodrigo Castro[1,2], Peter Fritzson[3], François Cellier[4],
Safa Motesharrei[5], Jorge Rivas[6]

[1]Department of Environmental Systems Science, ETH Zurich, Switzerland
[2]Department of Computer Science, University of Buenos Aires
and CIFASIS-CONICET, Argentina
[3]Department of Computer and Information Science, Linköping University,
SE-581 83 Linköping, Sweden
[4]Computer Science Department, ETH Zurich, Switzerland
[5]National Socio-Environmental Synthesis Center (SESYNC)
Annapolis, MD, 21401, USA
[6]Department of Political Science, University of Minnesota
Minneapolis, MN 55408, USA

rodrigo.castro@usys.ethz.ch, peter.fritzson@liu.se, fcellier@inf.ethz.ch,
ssm@umd.edu, jorgerodrigorivas@gmail.com

It is our predicament that we live in a finite world, and yet we behave as if it were infinite. Steady exponential material growth with no limits on resource consumption and population is the dominant conceptual model used by today's decision makers.

This is an approximation of reality that is no longer accurate and started to break down. The World3 model, originally developed in the 1970s, includes many rather detailed aspects of human society and its interaction with a resource-limited planet. However, World3 is a rather complex model.

Therefore it is valuable for pedagogical reasons to show how similar behavior can be also realized with models that are much simpler.

This paper presents a series of world models, starting with very simple exponential growth and predator-prey systems, then investigates a minimal human-nature model, Handy, and ends with a brief account of the World3 model.

For the first time, a simple human-nature interaction model is made available in Modelica that distinguishes between dynamics of Elite and Commoner social groups.

It is shown that Handy can reproduce rather complex behavior with a very simple model structure, as compared to that of world models like World3.

1D/2D Cellular Automata Modeling with Modelica

Victorino Sanz† Alfonso Urquia† Alberto Leva‡
† Dpto. Informática y Automática, ETSI Informática, UNED
Juan del Rosal, 16, 28040, Madrid, Spain
{vsanz,aurquia}@dia.uned.es
‡ Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano
Piazza Leonardo da Vinci 32, 20133 Milano, Italy
leva@elet.polimi.it

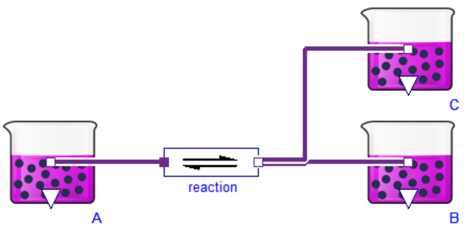
Cellular Automata (CA) can be used to describe dynamic phenomena dependent of the spatial coordinates. This approach exhibits two main advantages: CA models are conceptually simple and can be simulated very efficiently. A new Modelica library named *CellularAutomataLib* is presented. It facilitates describing one- and two-dimensional CA in Modelica, and interfacing these CA models with other Modelica models. Simulation performance and large model support have been highest priority in the design of the library. To achieve these goals, the CA internal description is programmed in C and it is consequently hidden to the modeling environment, which is released from the burden of causalizing and manipulating the millions of equations that typically compose CA models. The library architecture and use are discussed in this manuscript. Two examples illustrate the library use: heat diffusion on a chip and spread of an epidemic disease. *CellularAutomataLib* is freely available at `http://www.euclides.dia.uned.es`

Physiolibrary - Modelica library for Physiology

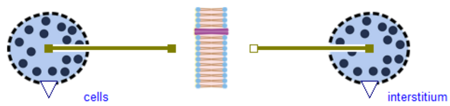
Marek Mateják*, Tomáš Kulhánek*, Jan Šilar*, Pavol Privitzer*,
Filip Ježek**, Jiří Kofránek*
*1st Faculty of Medicine, Charles University in Prague
U nemocnice 5, Prague 2, 128 53, Czech Republic
**Faculty of Electrical Engineering, Czech Technical University in Prague

marek@matfyz.cz

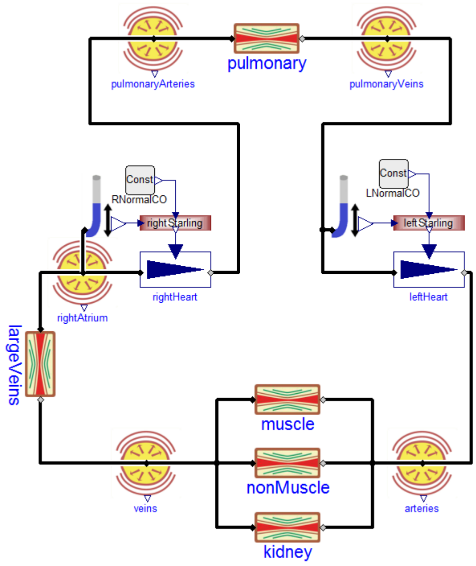
Physiolibrary is a free open-source Modelica library designed for modeling human physiology. It is accessible on the Modelica Libraries web page at https://www.modelica.org/libraries. This library contains basic physical laws governing human physiology, usable for cardiovascular circulation, metabolic processes, nutrient distribution, thermoregulation, gases transport, electrolyte regulation, water distribution, hormonal regulation and pharmacological regulation.



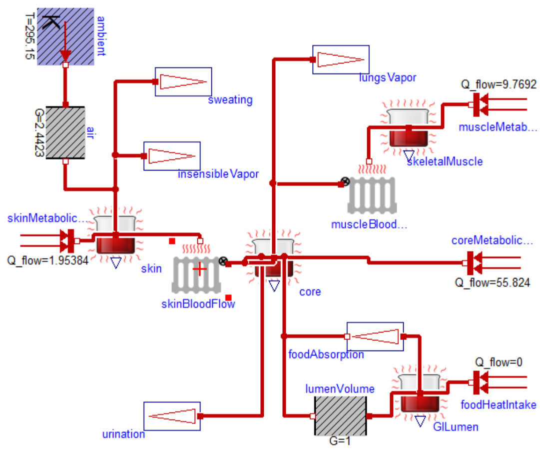
Chemical: substance, reaction, diffusion, clearance, degradation, stream, dilution,..



Osmotic: semi-permeable membrane, accumulation place for permeable liquid,..



Hydraulic: resistor, pump, elastic vessel, hydrostatic column, inertia, absorption, ..



Thermal: conductor, ideal radiator, heat accumulation place, stream, vaporization,..

References

[1] J. Kofránek, M. Mateják, P. Privitzer: HumMod - large scale physiological model in Modelica. 8th. International Modelica conference, Dresden, Germany, 2011.

Modelling of Electrical Power Systems with Dynamic Phasors in Modelica

<Tao Yang> <Serhiy Bozhko> <Greg Asher>
<University of Nottingham>
< University Park, Nottingham, NG7 2RD, UK >
<Tao.Yang, Serhiy.Bozhko, Greg.Asher@nottingham.ac.uk>

The More-Electric Aircraft (MEA) has been identified as the major trend of aircraft. Compared with conventional aircraft, the Electrical Power System (EPS) in MEA is a more complex system with increased numbers of power electronic converters (PEC) and electrical loads. In order to ensure the over-all performance of the EPS, it is essential to study the aircraft EPS at the system level. However, due to the switching behaviour of power electronic devices, it is very time-consuming and even impractical to simulate a large-scale EPS with some non-linear and time-varying models. In this paper, the Dynamic Phasor (DP) technique is introduced to model the EPS [1]. The DP is in nature some time-varying Fourier coefficients. Compared with traditional phasors, the DP can be used not only for steady-state studies, but also for transient studies.

A model library based on DP concept has been built. This paper summarizes the DP model library developed in The University of Nottingham for accelerated simulation studies of the aircraft EPS. The developed library includes Synchronous Generators, Controlled Rectifier Units (CRU) [2], Auto-Transformer Rectifier Units (ATRU) [3] etc. A twin-generator system based on the More-Open Electrical Technology (MOET) architecture is studied using developed DP models. The ABC models (Models in three-phase coordinates with switching behaviour) and DQ0 models (Averaging model in the dq frame) are used for comparative studies. The efficiency and the accuracy of the DP model are demonstrated through comparison of the simulation results with ABC and DQ0 models, under both normal and abnormal operation conditions.

References

[1] A. Emadi, "Modeling of power electronic loads in AC distribution systems using the generalized State-space averaging method," *Industrial Electronics, IEEE Transactions on*, vol. 51, pp. 992-1000, 2004.

[2] Y. Tao, S. Bozhko, and G. Asher, "Modeling of active front-end rectifiers using dynamic phasors," in *Industrial Electronics (ISIE), 2012 IEEE International Symposium on*, 2012, pp. 387-392.

[3] S. B. Tao Yang, Greg Asher, "Modeling of An 18-pulse Autotransformer Rectifier Unit with Dynamic Phasors," *SAE 2012*, 2012.

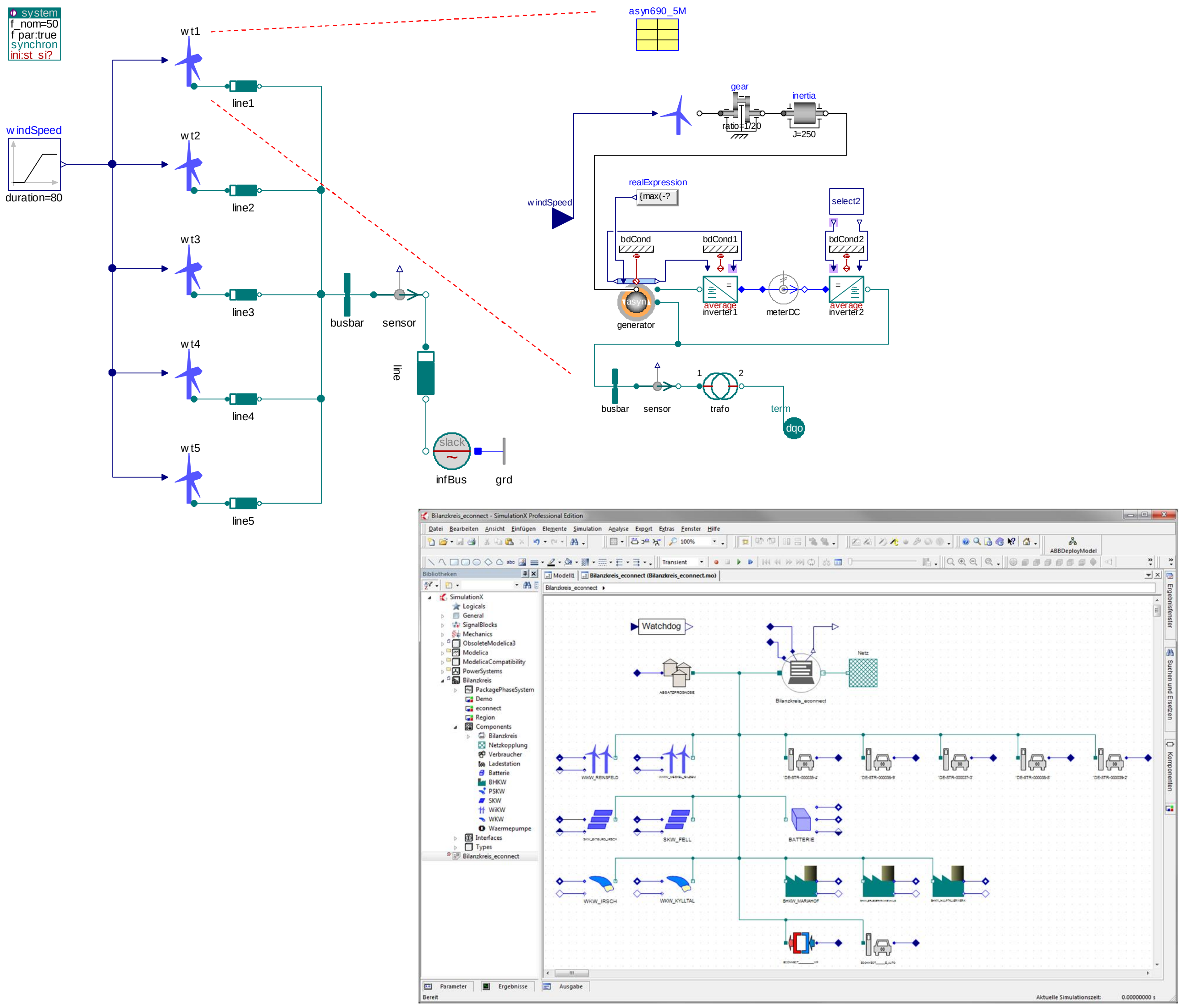
Flexible modeling of electrical power systems – the Modelica PowerSystems library

Rüdiger Franke, ABB, Germany – Ruediger.Franke@de.abb.com,
Hansjürg Wiesmann, Switzerland – HJ.Wiesmann@bluewin.ch

New trends such as renewable power, virtual power plants, electric mobility and smart grids raise the importance of electrical power systems. The systems are manifold, including e.g. DC, single- and multi-phase AC with fixed and variable frequency. Often times such systems cover other physical domains as well, such as rotational mechanics and thermo-fluid. Required system models range from simple flow calculations of active power to detailed transient and asymmetric studies of three-phase systems. Transformed modal coordinates play an important role for the treatment of three-phase AC systems.

The paper introduces the new Modelica PowerSystems library. It covers arbitrary phase systems in one modeling framework. Besides simple generic models that are valid with all phase systems, also large sets of detailed component models for DC and three-phase AC are included. The detailed component models have been ported from the former Spot library to PowerSystems.

The applications of one and the same PowerSystems library range from detailed transient drive train models up to online optimization models for the balancing of accounting grids.



Implementation of a Multi-Level Power Electronic Inverter Library in Modelica

C.I. Hill P. Giangrande C. Gerada S.V. Bozkho
The University of Nottingham
Aerospace Technology Centre, Innovation Park, Nottingham. NG7 2TU.
C.Hill@nottingham.ac.uk P.Giangrande@nottingham.ac.uk

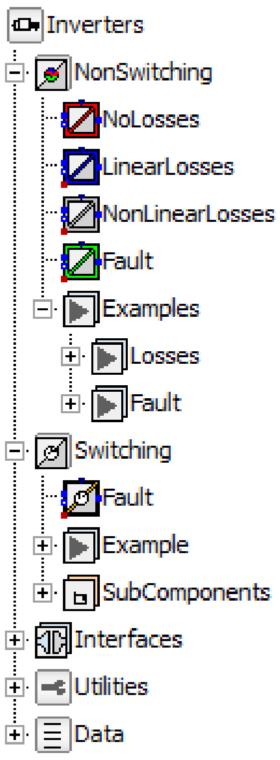


Figure 1: Structure of the presented library.

This paper presents a newly developed Power Electronic Inverter library implemented in Modelica. The library utilises a multi-level approach with increasing model complexity at progressively higher levels. All levels are fully interchangeable so as to provide a flexible library able to be utilised for investigation at single or multiple levels of complexity.

The library, simply named `Inverters`, provides the ability to include losses, analyse thermal response and introduce fault conditions into the modelling environment. It also gives the user multiple interchangeable models of the Power Electronic Inverter in order to allow analysis at multiple levels of complexity. The `Inverters` library presented here forms part of an overall `Actuator` library developed as part of Actuation 2015 [1]. For an overview of the entire `Actuator` library please see [2]. However the purpose of this paper is to give in-depth detail on the multi-level modelling of the Power Electronic Inverter.

The structure of the `Inverters` library is shown in Figure 1. A central feature of this `Inverters` library is that each modelling level is fully replaceable with one another. In order for each modelling level to be fully interchangeable a common interface is used for all 5 modelling levels. As a result the user can simply and easily investigate system responses under differing conditions and at varying levels of complexity

Within this interchangeable multi-level approach, there are two key attributes which are implemented into this new library. The first is the ability to include losses between the input and output of the Power Electronic Inverter. This is implemented so that the losses are included irrespective of the direction of power flow. Secondly, this library also provides the ability to trigger single or multiple open and short circuit faults within the Inverter. The library therefore provides an extremely useful tool able to compare system response under a variety of operational scenarios.

References

[1] Funded under the frame of the European Union Seventh Framework Program (FP7) http://www.actuation2015.eu/

[1] Van der Linden, F., Schlegel, C., Christmann, M., Regula, G., Hill, C.I., Giangrande, P., Mare, J.C., Egaña, I. "Implementation of a Modelica Library for Simulation of Electromechanical Actuators for Aircraft and Helicopters." *Proceedings of the 10th International Modelica Conference*. 2014.

Mixed phasor and time domain modelling of AC networks with changeover management

Hakan Dogan Parildar* Alberto Leva†
Via Ponzio 34/5, 20133, Milano, Italy
parildar@sabanciuniv.edu, alberto.leva@polimi.it

Simulation studies on AC electric networks may comprehend periods of quasi-stationary operation and rapid transients. The adoption of a phasor-based approach results in high simulation efficiency, but is limited to the first of the two situations above, while for the second, time domain models are required. For system studies where both situations have to be simulated, a modelling paradigm is thus required that can join the two approaches in all the described components, and by which the simulator of an entire network can be endowed with the capability of moving back and forth from a phasor to a time domain system description automatically, taking care of the proper re-initialisations when necessary.

In this paper we propose a possible solution, structured into a *component-level* part and a *system-level* one. This allows to preserve the object-oriented nature of the constructed models, even though the automatic changeover from phasors to time domain mode is a decision to be taken at the overall system level. For that particular purpose, we introduce a mechanism based on continuous-time filters, that allows to exploit the capabilities of variable-step solvers in a view to efficiency, and limits the number of changeover-generated state events to the bare necessary.

As a result of the design sketched out above, the additional effort required on the part of the analyst for using the proposed mixed modelling paradigm is reduced to a minimum, and – which is equally important – requires a really minimal (if any) and easy to understand knowledge of the underlying mechanism.

To prove the viability and usefulness of the proposed ideas, we put them to work, and as a consequence we present the first *nucleus* of a Modelica library designed along them. We also show some simulation examples to support the validity and practical convenience of the proposal.

*Former graduated student at the Dipartimento di Elettronica, Informazione e Bioingegneria.
†Corresponding author.

`impact` – A Modelica® Package Manager

Michael Tiller
Xogeny Inc., USA
`michael.tiller@xogeny.com`

Dietmar Winkler
Telemark University College, Norway
`dietmar.winkler@hit.no`

To manage complexity, modern programming languages use organizational units to group code related by some common purpose. Depending on the programming language, these units might be called libraries, packages or modules. But they all attempt to encapsulate functionality to promote modular code and reusability. For the remainder of this paper, we will simply refer to these organizational units as *packages* (as they are called in Modelica).

Also common to many modern programming languages are tools to manage these packages. These tools are generally called *package managers* and they allow developers to quickly "fetch" any packages they may need for a given project. The main functions of package managers are to allow developers to search, install, update and uninstall packages with a simple command-line or graphical interface. In the Java world, the most common package manager is `maven`. For Python, tools like `easy_install`[1] and `pip`[2] are used for managing packages. For client-side web development, `bower` is used. For server-side JavaScript, the tool of choice is `npm`[3]. For compiled languages, these package managers often include some additional build functionality as well.

This paper introduces *`impact`*, a package manager for Modelica. Using `impact`, Modelica users and developers can quickly search for, install and update Modelica libraries. In this paper, we will discuss the functionality provided by `impact`. In addition, we will discuss how the functionality was implemented. As part of this we will discuss the importance of collaborative platforms, like GitHub[4] in our case, for providing a means for collecting, curating and distributing packages within a community of developers.

The `impact` package manager is provided to the Modelica community as a free, open-source tool. Furthermore, the protocols involved are all documented and we encourage tool vendors to integrate them into their own tools so they can provide the same searching, updating and installation capabilities that the command-line tool provides.

References

[1] easy_install. *Easily download, build, install, upgrade, and uninstall Python packages.* 2014. URL: `https://pypi.python.org/pypi/setuptools`.

[2] pip. *A tool for installing and managing Python packages.* 2014. URL: `http://www.pip-installer.org`.

[3] npm. *Node Packaged Modules.* 2014. URL: `https://npmjs.org/`.

[4] GitHub. *Build software better, together.* 2014. URL: `https://github.com/`.

MoUnit — A Framework for Automatic Modelica Model Testing

Roland Samlaus[1] Mareike Strach[1] Claudio Hillmann[1] Peter Fritzson[2]

Fraunhofer IWES, Turbine Simulation, Software Development, and Aerodynamics[1]
Department of Computer and Information Science Linköping University[2]

A vital part in development of physical models, i.e., mathematical models of physical system behavior, is testing whether the simulation results match the developer's expectations and physical laws. Creation and automatic execution of tests need to be easy to be accepted by the user. Currently, testing is mostly performed manually by regression testing and investigation of result plots. Furthermore, comparisons between different tools can be cumbersome due to different output formats. In this paper, the test framework MoUnit is introduced for automatic testing of Modelica models through unit testing. MoUnit allows comparison of Modelica simulation results with reference data, where both reference data and simulation results can originate from different simulation tools and/or Modelica compilers. The presented test framework MoUnit brings the widespread approach of unit testing from software development into practice also for physical modeling. The testing strategy that is used within the Modelica IDE OneModelica from which the requirements for MoUnit arose, is introduced using an example of linear water wave models. The implementation and features of MoUnit are described and its flexibility is exhibited through two test cases. It is outlined, how MoUnit is integrated into OneModelica and how the tests can be automated within continuous build environments.

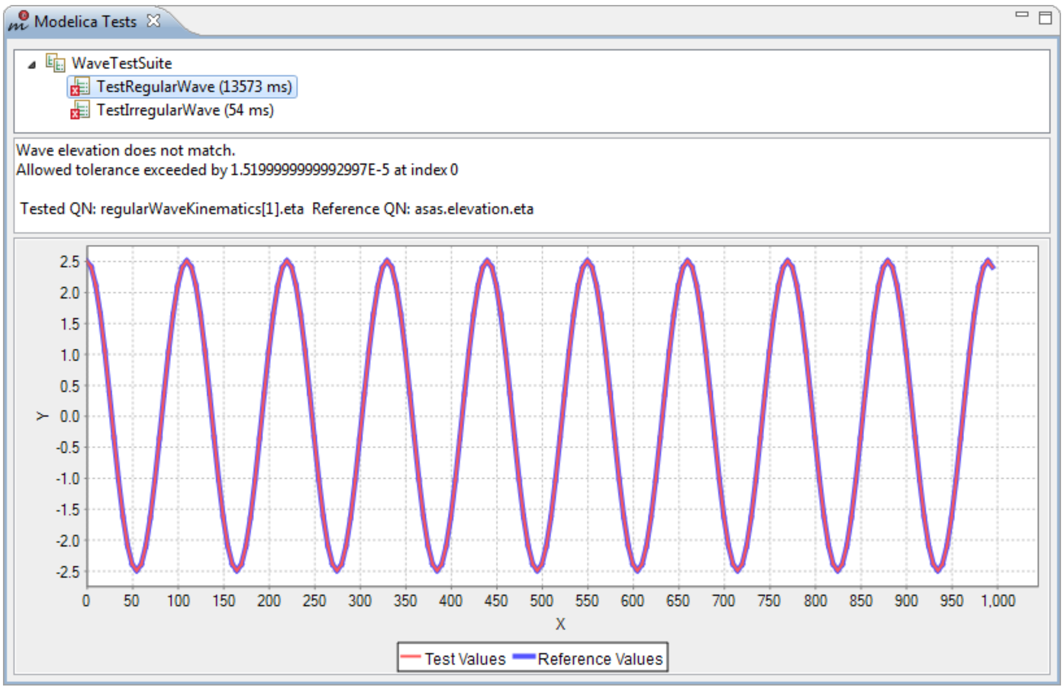


Figure 1: Result of a MoUnit test case

Modeling Parameter Sensitivities via Equation-based Algorithmic Differentiation Techniques
The ADMSL.Electrical.Analog Library

Atiyah Elsheikh
Austrian Institute of Technology, Vienna, Austria
Atiyah.Elsheikh@ait.ac.at

Figure 1 presents a standard electrical circuit example, the `ChuaCircuit` model from the standard Modelica library `Modelica.Electrical.Analog.Example`. By simulating this example, parameter sensitivities are additionally evaluated. The curves in the upper part and the lower part correspond to the parameter sensitivities of

- the current at all components (i.e. `L.i,Ro.i,G.i,C1.i,C2.i,Nr.i,Gr.i`) w.r.t. the inductance parameter `L.L` and
- the voltage at the `Inductor L` w.r.t. all parameters (i.e. `L.L,Ro.R,G.G, C2.C, etc.`) respectively

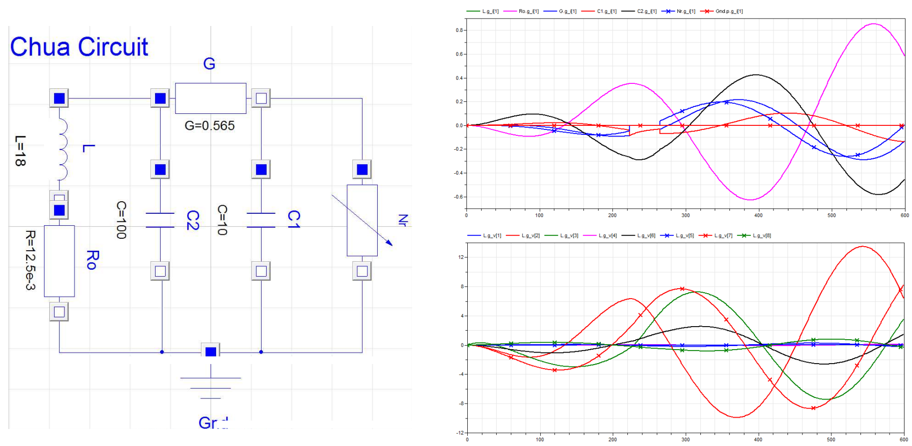


Figure 1: The Chua Circuit standard example and simulation of parameter sensitivities

This example is within the ADMSL library [1] (Algorithmically Differentiated Modelica Standard Library). The ADMSL library is an Algorithmic Differentiation (AD) version of the standard library `Modelica.Electrical.Analog.Basic`. It has an identical structure to the original standard library, but it enhances the components with additional declaration and equations for describing parameter sensitivities.

The algorithmically differentiated library ADMSL serves as an illustrative example for applying novel equation-based AD techniques on Modelica libraries. It illustrates the basic steps required to systematically transform a Modelica library to another library that additionally describes models parameter sensitivities. The produced library remains with the same structure and the underlying models keep the same interface and outlook.

References

[1] A. Elsheikh. The ADMSL library. https://github.com/AIT-CES-LAB/ADMSL.

Modelica Based Parser Generator with Good Error Handling

Arunkumar Palanisamy[1], Adrian Pop[1], Martin Sjölund[1], Peter Fritzson[1]

[1]PELAB – Programming Environment Laboratory
Department of Computer and Information Science
Linköping University, SE-581 83 Linköping, Sweden
{arunkumar.palanisamy, adrian.pop, martin.sjolund, peter.fritzson}@liu.se

This paper describes the new OpenModelica Compiler-Compiler (OMCC) including a parser generator, OMCCp which is based on an LALR parser generator extended with advanced error handling facilities. It is implemented in the MetaModelica language with parsing tables generated by the tools Flex and Bison. It is integrated with the MetaModelica semantics specification language, based on operational semantics for generating executable compiler and interpreter modules.

The OMCCp parser generating part of OMCC has been used for the full Modelica language grammar as well as for the language extensions of MetaModelica, ParModelica, and Optimization specifications. It has been successfully tested on MSL models and on the OpenModelica test suite comprising more than 1200 example models. The generated parsers have reasonable performance compared to most other parser generators.

Keywords: Modelica, MetaModelica, Parsing, Compilers, Error Handling, Flex, Bison, ParModelica, Optimization, OMCCp

Nonlinear inverse models for the control of satellites with flexible structures

Matthias J. Reiner Johann Bals
German Aerospace Center (DLR)
Institute of System Dynamics and Control
82234 Wessling, Germany
{Matthias.Reiner, Johann.Bals}@DLR.de

Nonlinear inverse dynamic models can be utilized in various parts of advanced model-based control system design: reference trajectory optimization, feed-forward control and feedback linearization [3]. In this paper, a new synthesis approach for nonlinear inverse dynamic models of satellites with flexible structures is presented. For satellite configurations with unstable zero dynamics, a stable inverse model approximation is proposed which has been successfully applied to robots with flexible bodies [2, 1].

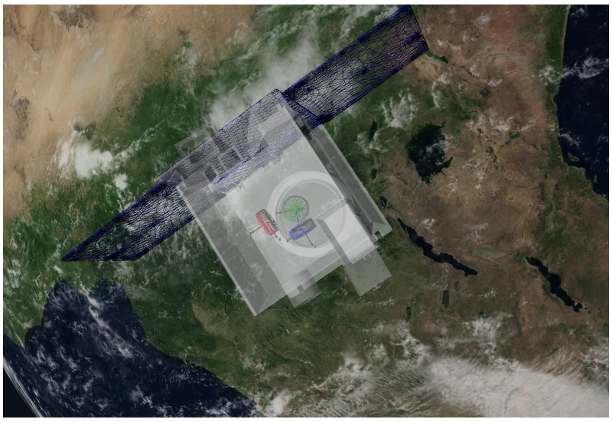


Figure 1: Animation of a satellite with the DLR Space Systems Library.

This inverse modeling approach is part of the newly developed DLR Space Systems Library for object-oriented modeling and simulation of satellites and launchers in a detailed space environment. For satellites with flexible structures, the library provides models for normal simulation mode and the necessary tools to directly generate approximate inverse models.

In this paper, trajectory optimization is shown to be an important use case for inverse dynamic models. By inversion based reformulation of the trajectory optimization problem, the optimal reference motion of the control system can be determined in a reliable and efficient way.

References

[1] M. Reiner. *Modellierung und Steuerung eines strukturelastischen Roboters*. PhD thesis, Technische Universität München, 2011.

[2] M. Reiner, M. Otter, and H. Ulbrich. Modeling and feed-forward control of structural elastic robots. *AIP Conf. Proceedings of the Int. Con. of Numerical and Analysis and Applied Mathematics*, pages 378–381, 2010.

[3] M. Thümmel, G. Looye, M. Kurze, M. Otter, and J. Bals. Nonlinear inverse models for control. *Proceedings of the 4th International Modelica Conference,Hamburg, March 7-8*, 2005.

Modelica Stage Separation Dynamics Modeling for End-to-End Launch Vehicle Trajectory Simulations

Paul Acquatella B., and Matthias J. Reiner
German Aerospace Center (DLR), Institute of System Dynamics and Control
82234 Wessling, Germany
paul.acquatella@dlr.de · matthias.reiner@dlr.de

Stage separation dynamics modeling is a critical capability of future launchers preparatory studies. The development of stage separation frameworks integrable in end-to-end launch vehicle trajectory simulations have been presented in the relevant literature but none profiting from the object-oriented and equation-based acausal modeling properties of MODELICA [1, 2]. The objective of this paper is therefore to present such an approach to this problematic. Based on the *Constraint Force Equation* (CFE) methodology [3, 4, 5], two case studies to evaluate the proposed approach are considered. Results demonstrate that the approach corresponds very well with the physics behind separation. In addition, we found easiness of implementation of the method within a single environment such as DYMOLA, demonstrating the benefits of an integrated approach.

References

[1] Elmqvist, H., Mattson, S.E., and Otter, M. *Modelica - An International Effort to Design an Object-Oriented Modeling Language.* Summer Computer Simulation Conference, pp. 333-339, 1998, ISBN 1-56555-149-4.

[2] Mattson, S.E., Elmqvist, H., and Otter, M. *Physical system modeling with Modelica.* Control Engineering Practice, v. 6, pp. 501-510, 1998.

[3] Toniolo, M., Tartabini, P.V., and Pamadi, B.N. *Constraint Force Equation Methodology for Modeling Multi-Body Stage Separation Dynamics.* 46th AIAA Aerospace Sciences Meeting and Exhibit, 2008, 10.2514/6.2008-219.

[4] Tartabini, P.V., Roithmayr, C.M., Toniolo, M.D., Karlgaard, C.D., and Pamadi, B. N. *Modeling Multibody Stage Separation Dynamics Using Constraint Force Equation Methodology.* Journal of Spacecraft and Rockets, Vol. 48, No. 4, July-August 2011, pp 573-583.

[5] Pamadi, B.N., Tartabini, P.V., Toniolo, M.D., Roithmayr, C.M., Karlgaard, C.D., and Samareh, J.A. *Application of Constraint Force Equation Methodology for Launch Vehicle Stage Separation.* Journal of Spacecraft and Rockets, Vol. 50, No. 1, January-February 2013, pp 191-205.

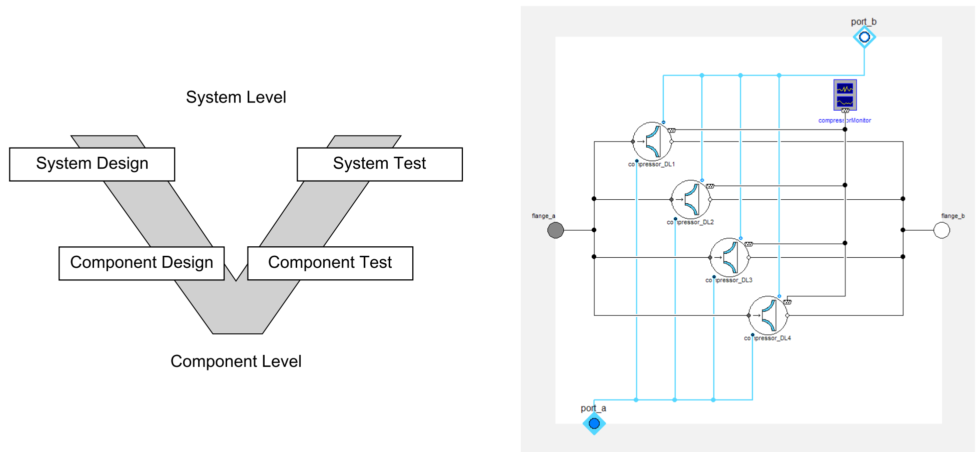
A Modelica Library for Scalable Modelling of Aircraft Environmental Control Systems

Philip Jordan   Gerhard Schmitz
Hamburg University of Technology, Institute of Thermo-Fluid Dynamics
Denickestr. 17, 21075 Hamburg

In this paper, the Modelica library developed in the Clean Sky project TEMPO [1] is presented. The aim of the library is to support scalable system models for aircraft environmental control systems, which can be modified in detail and characteristic to be used during different phases of the design cycle without the need of rebuilding the system model or switching to another software tool.

The design process of complex technical systems such as aircraft environmental control and cooling systems can be broken down into characteristic design phases (figure 1a). Depending on the design phase, models of the system under development may differ in structure and complexity, as they are designed to answer design phase specific questions. The library presented in this paper aims at using the object-oriented features of the Modelica language to integrate multiple layers of models. Each layer corresponds to a set of models designed for use in a particular design phase. The design phase-specific models are integrated such that a single system model that has been assembled from available library component models can be “scaled”, i.e. reused, for a simulation activity of a different phase of the system design cycle.

The library structure and the chosen container model approach (figure 1b) to integrate the different detail levels is outlined. The functionality of the library is then demonstrated at the example of a generic aircraft environmental control system architecture.



(a) (b)

Figure 1: (a): V-diagram showing the different phases of a system design cycle. (b): Diagram layer of a scalable component model using the container approach

References

[1] Clean Sky JTI, “Systems for green operations technology demonstrator overview.” `http://www.cleansky.eu/content/project/system-green-operations`, accessed October 2013.

Model-Based Energy Recuperation of Multi-Axis Machines

Tamás Juhász, Matthias Kennel, Marco Franke, Ulrich Schmucker
Fraunhofer Institute for Factory Operation and Automation IFF
Sandtorstr. 22, 39106 Magdeburg, Germany
{tamas.juhasz, matthias.kennel, marco.franke, ulrich.schmucker}@iff.fraunhofer.de

The peak power consumption of multi-axis production machinery (e.g. industrial robots) is determined by the forces during acceleration phases of continuous movements. The required high electric currents represent a cost factor in terms of the mains power supply. In this paper a new Modelica-based method is presented to save the mechanical braking energy of production machinery into a local flywheel-based energy recuperation system (ERS) for later utilization. An ERS has twofold advantages: on the one side it reduces the apparent power peaks from the mains power system. On the other side the overall energy consumption can also be reduced therewith. However, a mechanical ERS with a flywheel needs to be controlled in advance, as its internal inertia and the switched magnetic field introduce some dead time in the process. Therefore, the here presented approach uses an interdisciplinary Modelica model of the machinery to compute future power requirements prior execution of new movements. It is assumed that the machine program is known upfront in a textual form. The movement commands must be carried out with the virtual machine model first. The simulation computes the energy demand, according to which the stored amount of energy within the ERS (i.e. the angular velocity of its flywheel) must be controlled. The here computed reference angular velocity signal is put later also to the real controller, where the physical ERS is attached to as an extra motor. This paper presents the methodology along an example of a 3-axis robotic manipulator that is installed in the VDTC building of the Fraunhofer IFF, Magdeburg. This specific example has been used to validate the concept: 12% less power-consumption and 10% less power peaks were achieved during the operation.

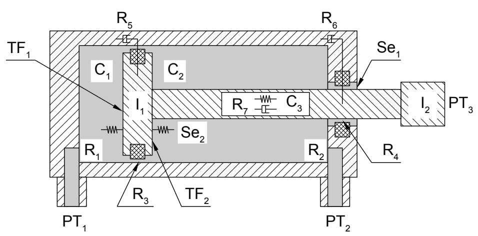
A Generalized Power-Based Modelica Library with Application to an Industrial Hydraulic Plant

Ilja Alkov, Robin Diekmann, Dirk Weidemann
Institute of System Dynamics and Mechatronics
University of Applied Sciences Bielefeld
Wilhelm-Bertelsmann-Str. 10, 33602 Bielefeld, Germany
{ilja.alkov, robin.diekmann, dirk.weidemann}@fh-bielefeld.de

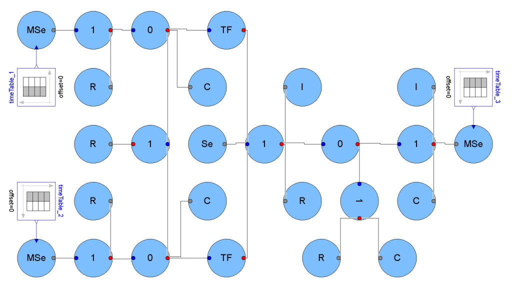
The generalized power-based approach is an efficient modeling formalism based on the idea that in each technically relevant discipline, power is represented by the product of an effort value and a flow value. By the generalized power definition, unified modeling elements are conceptualized applicable to the direct multidisciplinary modeling of complex systems. Consequently, the modeling procedure consists of the characterization of domain specific processes corresponding to their unified complements and interconnecting these complements to the model of complete system according to its structure.

The Bond Graph (BG) formalism provides a multidisciplinary, generalized power-based approach to the modeling and also graphical model representation of dynamic systems. By the object-oriented nature, BG possess the according advantages. The Modelica language and the BG formalism are partly closely related modeling methodologies. Hence, an implementation of the BG formalism in Modelica is considered in this contribution. The proposed implementation attempts to take advantages of both modeling concepts.

The corresponding developed library BondGraph available online at the official Modelica web page [1] is discussed in detail. It allows graphical modeling according to the bond graph formalism, and contains common bond graph elements, as well as specific nonlinear elements, especially related to hydraulic effects. For these, attention has been paid to their numerically stable computation. Furthermore, several composed models are provided, such as switching valves, pipes, cylinders, etc. A combination with blocks of the Modelica Standard Library is possible. The application of BondGraph to an industrial plant is described to demonstrate its capabilities.



(a) Iconic model of a hydraulic cylinder.



(b) According Bond Graph representation of the cylinder using the BondGraph library in Dymola.

References

[1] I. Alkov and R. Diekmann. (2013) BondGraph library. [Online]. Available: https://www.modelica.org/libraries

Physical Design of Hydraulic Valves in Modelica

Sureshkumar Chandrasekar   Hubertus Tummescheit
Modelon Inc., Hartford, CT, USA
chandrasekar@modelon.com   hubertus.tummescheit@modelon.com

Abstract

This paper focuses on the physical design of hydraulic relief and servo valves and its applications. Specifically, this paper serves to illustrate how the physical design parameters of hydraulic components can be incorporated into system modeling and their effect on the system dynamics and stability characteristics. Detailed physical models of a relief valve and a servo valve developed using the Hydraulics Library® are discussed in this paper with particular emphasis on the effect of design parameters on the stability characteristics. A simple design of experiment (DoE) to illustrate robust design methods for hydraulic system design is also shown with the use of the FMI Toolbox (FMIT) for MATLAB®. Furthermore with the help of these two valve models, we seek to bring to the attention of the community, a limitation in open loop controls analysis in an acausal modeling environment where the feedback loops are embedded in the physics of the model.

References

[1] Merritt, H.E., "Hydraulic Control Systems", *John Wiley Sons, Inc.*, 1967.

[2] Edwards J.W., "Analysis of an Electro-Hydraulic Aircraft Control-Surface Servo and Comparison with Test Results", *NASA Technical Note NASA TN D-6928*, 1972.

[3] Eaton Corp., "Product Literature Library Proportional Directional Valves without Feedback, (V-VLPO-MC007-E), Proportional Valves with Feedback, (V-VLPO-MC005-E)", *Accessed from http://hydraulics.eaton.com/products/valves_ proportional_ directional_ controls.htm*, in Nov 2013.

[4] Tchkalov V., Miller S., "Parameterization of Directional and Proportional Valves in Sim-Hydraulics", *Accessed from http://www.mathworks.com/matlabcentral/file exchange/27260* in Nov 2013.

[5] Modelon AB, "FMI Toolbox for MATLAB", *Accessed from http://www.modelon.com/products/fmi-toolbox-for-matlab/* in Nov 2013.

[6] Henningson M., Akesson J., Tummescheit H., "FMI-Based Tools for Robust Design of Dynamical Systems", *Proceedings of the 10th International Modelica Conference*, Lund 2014.

Exploiting Actuator Limits with Feedforward Control based on Inverse Models

Manuel Gräber
TLK-Thermo GmbH
Hans-Sommer-Str. 5, 38106 Braunschweig, Germany
m.graeber@tlk-thermo.de

Feedforward control based on inverse dynamic plant models (linear or nonlinear) is a suitable method to enhance set-point tracking performance of control systems. Modelica as equation-based modeling language provides a powerful possibility to automatically generate inverse dynamic models. In many cases inverse models can directly be derived from forward models. Details about model inversion with Modelica can be found in [1].

In reality actuators always have limits, but limiting functions can not be inverted. A common approach to handle this issue is to invert the unlimited plant model and detune the feedforward filter in order to stay always in between the actuator limits. This approach causes a loss in performance for rapid set-point changes, because the actuator range is not entirely used. In this article a rather simple but powerful method is presented, which overcomes this performance issue for many types of plant models. Actuator limits are fully exploited, and the obtained trajectories are close to optimal ones. Simulation and measurement results (see Figure 1) demonstrate the usability of the proposed feedforward structure.

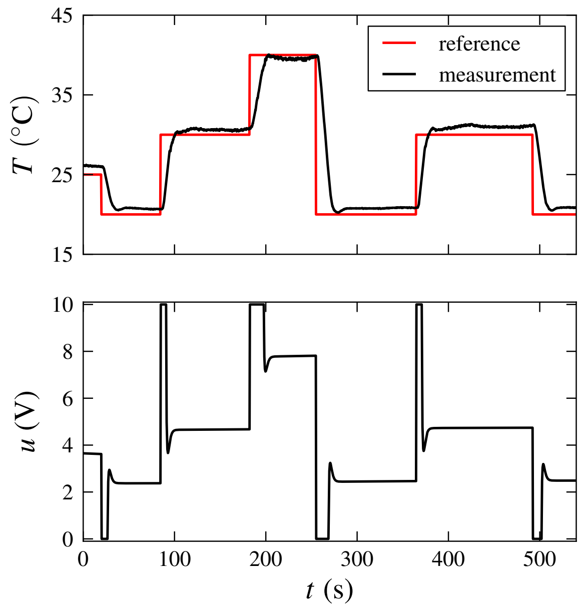


Figure 1: Measurement results from feedforward control of a liquid heater. Using an inverse plant model and a reference trajectory for the temperature T, almost time-optimal plant input values (u) are computed.

References

[1] Gertjan Looye, Michael Thümmel, Matthias Kurze, Martin Otter, and Johann Bals. Nonlinear Inverse Models for Control. In *Proc. of the 4th International Modelica Conference*, Hamburg, 2005.

An FMI-based Framework for State and Parameter Estimation

Marco Bonvini, Michael Wetter, Michael D. Sohn
Simulation Research Group, Lawrence Berkeley National Laboratory
1 Cyclotron Road, 94720, Berkeley, CA
{MBonvini, MWetter, MDSohn}@lbl.gov

In many applications, after the system has been designed, controls and/or fault detection and diagnostics (FDD) algorithms are developed and deployed. These techniques should be able to leverage the models developed during the earlier design stages, thereby increasing the productivity of the overall product development. Advanced control (such as adaptive control or model predictive control) and FDD techniques require an enhanced knowledge of the system state. For example, the flight controller of an airplane should try to estimate the real velocity and position of the aircraft while compensating for measurements errors and sensor noises. When dealing with dynamic system, having an enhanced knowledge about the system state means estimating its state variables with associated error bounds.

This paper proposes a solution for creating a model-based state estimator for dynamic systems described using the FMI standard. This work extends the capabilities of any modeling framework compliant with the FMI standard version 1.0. The FMI is a standard that allows to embed a simulation model within a unified interface in order to couple simulation models developed using different simulation programs. Although the FMI standard has been created mainly for co-simulation, we leverage this standard for providing an algorithm that is compatible with a large number of simulation and modeling platforms, including Modelica-based ones.

The state estimation technique used in this work is the unscented Kalman filter (UKF) [1, 2]. The UKF is able to deal with nonlinear systems and it just requires to perform function evaluations of the model in order to compute the evolution of its state variables and the value of its outputs. The UKF has less requirements about the knowledge of the model with respect to other nonlinear state estimation techniques. For example the extended Kalman filter needs to linearize the model [3]. The computational performances of the UKF are modest with respect to other Monte Carlo based techniques (like particle filters [4]), enabling its use for real-time applications.

References

[1] S. J. Julier and J. K. Uhlmann. A general method for approximating nonlinear transformations of probability distributions. *Robotics Research Group Technical Report, Department of Engineering Science, University of Oxford*, pages 1–27, November 1996.

[2] S.J. Julier. The scaled unscented transformation. In *American Control Conference, 2002. Proceedings of the 2002*, volume 6, pages 4555–4559 vol.6, 2002.

[3] Simon S Haykin et al. *Kalman filtering and neural networks*. Wiley Online Library, 2001.

[4] D. Crisan and Arnaud Doucet. A survey of convergence results on particle filtering methods for practitioners. *Signal Processing, IEEE Transactions on*, 50(3):736–746, 2002.

Grey-Box Building Models for Model Order Reduction and Control

Roel De Coninck[a,b], Fredrik Magnusson[c], Johan Åkesson[c,d], Lieve Helsen[b]
[a]3E nv., 1000 Brussels, Belgium,
[b]KU Leuven, Department of Mechanical Engineering, 3001 Heverlee, Belgium,
[c]Department of Automatic Control, Lund University, SE-221 00 Lund, Sweden,
[d]Modelon AB, Ideon Science Park, SE-223 70 Lund, Sweden

Grey-Box modelling is considered as a strong framework for the creation of low-order models for analysis and control of monitored buildings. This paper presents an approach to obtain useful grey-box models in a largely automated way.

The first step is the creation of a building library with many potential model candidates. The Modelica package *FastBuildings* contains low-order models for thermal zones, HVAC, users, single and multi-zone buildings.

Next, a toolbox is presented that largely automates the parameter estimation of the *FastBuildings* models. It is implemented as a Python module that wraps the functionality of *JModelica.org* and presents the user a high-level interface for all common operations like data handling, model selection, parameter estimation and validation. A high-level overview of the toolbox is shown in Figure 1.

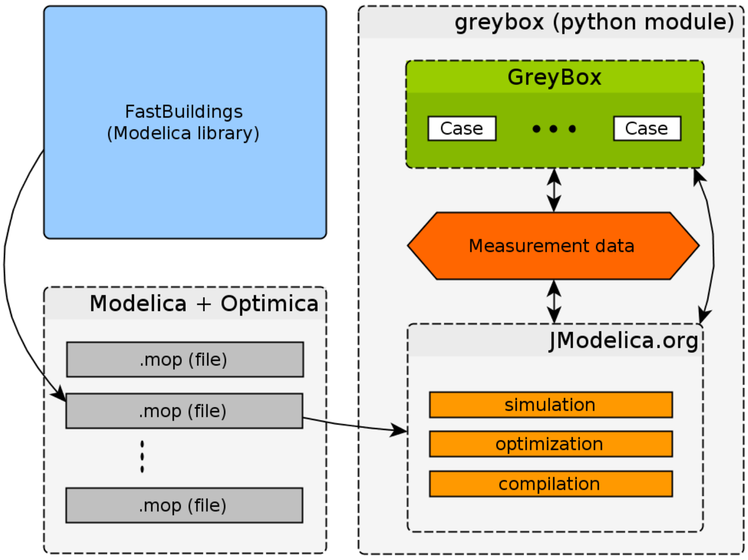


Figure 1: Overview of the grey-box buildings toolbox

The use of a gradient-based method allows an efficient numerical solution of the estimation problems. Specific attention is paid to robustness and ease-of-use. A Latin hypercube sampling of the parameter search space overcomes issues related to the non-convexity of the optimization problem.

The toolbox is applied to a model order reduction case study for a single-family dwelling. The selected model has 11 parameters and is able to predict the indoor temperature in an open-loop simulation (with a priori knowledge about weather and electricity consumption) with an RMSE of 0.16 K.

Interfacing Models for Thermal Separation Processes with Fluid Property Data from External Sources

Kai Wellner[1] Carsten Trapp[2] Gerhard Schmitz[1] Francesco Casella[3]
[1]Hamburg University of Technology, Institute of Thermo-Fluid Dynamics
Denickestraße 17, 21075 Hamburg, Germany
[2]Delft University of Technology, Propulsion and Power
Kluyverweg 1, 2629 HS Delft, The Netherlands
[3]Politecnico di Milano, Dipartimento di Elettronica, Informazione e Bioingegneria
Via Ponzio 34/5, 20133 Milano, Italy
{kai.wellner,schmitz}@tuhh.de

So far, when modelling processes that demand for multi-phase and multi-component fluid property data, the user has to implement the required media models in the Modelica language as these types of fluids are not supported by Modelica.Media. This paper presents a first approach on how to implement fluid property data in process models of an existing library from external sources and highlights which problems have to be overcome. Furthermore, it provides recommendations for the design of an efficient and user-friendly interface to external media packages.

Two approaches on the use of external property packages for the computation of fluid data interfaced to Modelica process models are compared to each other. With an exemplary column model from the `ThermalSeparation` library [1] using an explicit formulation of the balance equations the use of partial derivatives is avoided. This model is in terms of computational speed favourable in comparison to a model with an implicit formulation of the balance equations as illustrated in figure 1. However, especially when utilizing existing model libraries rewriting balance equations is not convenient or feasible. The paper shows that in this case the external property package has to provide partial derivatives of the fluid properties with respect to the system states.

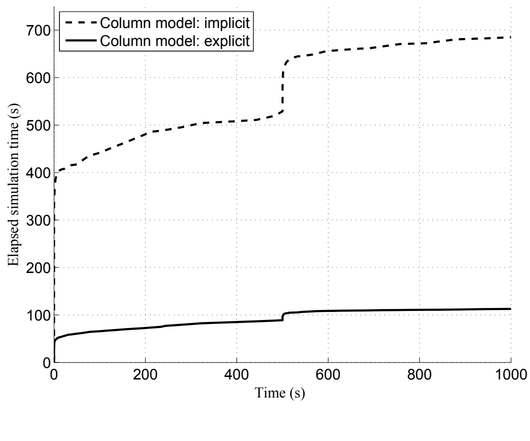


Figure 1: Comparison of simulation times.

References

[1] Dietl, K.: Equation-Based Object-Oriented Modelling of Dynamic Absorption and Rectification Processes, Hamburg University of Technology, Institute of Thermo-Fluid Dynamics, Hamburg, 2012, ISBN 978-3-8439-0778-1.

Development of a Real-Time Fuel Processor Model for HIL Simulation

Karin Fröjd[1], Karin Axelsson[2], Ivar Torstensson[1], Erik Åberg[1], Erik Osvaldsson[2], Gregor Dolanc[3], Bostjan Pregelj[3], Jonas Eborn[1], Jens Pålsson[1]

[1] Modelon AB, Ideon Science Park, Scheelev. 17, 223 70 Lund, Sweden

[2] PowerCell AB, Ruskvädersgatan 12, 418 34 Göteborg, Sweden

[3]J. Stefan Institute, Jamova cesta 39, 1000 Ljubljana, Slovenia

Karin.Frojd@modelon.com, Karin.Axelsson@powercell.se

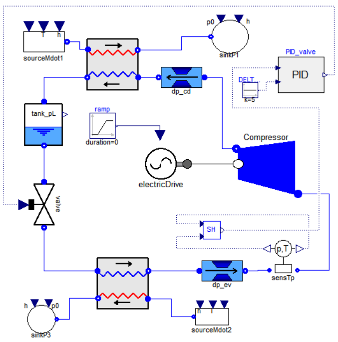
In this article a real-time model for dynamic simulation of a fuel processor is presented. The model is intended for HIL testing of the PLC for a truck Auxiliary Power Unit (APU) system.

The APU comprises a PEM fuel cell and fuel processor to enable direct utilization of on-board diesel. The system is under development in FCGEN, an EU project under the FP7 program FCH JU **Error! Reference source not found.**. One critical challenge is to design the control system (PLC) to ensure failsafe and environmental friendly startup and operation. The startup phase of the fuel processor is the most critical part, since it is a highly dynamic process involving several complex reactors. It is advantageous to verify the control system before the fuel processor is assembled to avoid possible breakage of components. Such verification can be done with a real-time model representing the physical system. In this study such a model is created using Modelica and Dymola. It is shown that it is possible to load and execute a real-time Modelica model capable of realistically mimicking the system response on a HIL platform. The model runs in real time using a first order explicit (Euler) solver with a time step size of 25 ms.

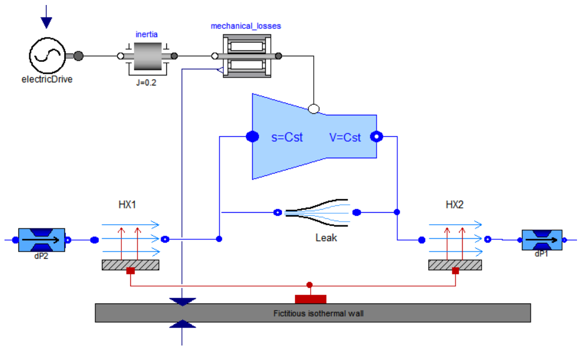
ThermoCycle: A Modelica library for the simulation of thermodynamic systems

Sylvain Quoilin[1], Adriano Desideri[1], Jorrit Wronski[2], Ian Bell[1] and Vincent Lemort[1]
[1]University of Liège, Energy Systems Research Unit
Bâtiment B49, Chemin des Chevreuils 7, 4000 Liège, Belgium
{squoilin,adesideri,ian.bell,vincent.lemort}@ulg.ac.be
[2]Department of Mechanical Engineering, Technical University of Denmark
Nils Koppels Allé 403, 2800 Kgs. Lyngby, Denmark
jowr@mek.dtu.dk

This paper presents the results of an on-going project to develop ThermoCycle, an open Modelica library for the simulation of low-capacity thermodynamic cycles and thermal systems. Special attention is paid to robustness and simulation speed since dynamic simulations are often limited by numerical constraints and failures, either during initialization or during integration. Furthermore, the use of complex equations of state (EOS) to compute thermodynamic properties significantly decreases the simulation speed.



(a) Model of an R407c heat pump



(b) Semi-empirical compressor model

Figure 1: Components of the ThermoCycle library

The currently available Modelica solutions for computing organic fluid properties are limited to non-open-source databases. In order to propose a fully open-source tool, ThermoCycle has been coupled to CoolProp, an open-source library developed at the University of Liège.

Furthermore, to enhance the performance and the robustness of the ThermoCycle library, different numerical methods have been implemented and tested. Some are implemented at the Modelica level while others require a modification of the thermodynamic properties of the working fluid and are therefore implemented into CoolProp. It should also be noted that some of these methods have already been proposed in the literature, while some others are new. In this paper, the approach adopted in the library to overcome the challenging task of dynamic modelling of thermodynamic cycles is presented and discussed.

An Operational Semantics for Hybrid Systems involving Behavioral Abstraction

Simon Bliudze
École Polytechnique Fédérale de Lausanne
INJ Building, Station 14
1015 Lausanne, Switzerland
simon.bliudze@epfl.ch

Sébastien Furic
LMS Imagine S.A.
7 place des Minimes
42300 Roanne, France
sebastien.furic@lmsintl.com

Abstract

We discuss the challenges of building a simulation framework for hybrid systems, in particular the well-known Zeno effect and correct composition of models idealised by abstracting irrelevant behavioural details (e.g. the bounce dynamics of a bouncing ball or the process of fuse melting in an electrical circuit). We argue that the cornerstone of addressing these challenges is the definition of a semantic framework with an appropriate underlying model of time.

Using two simple examples, we illustrate the properties of such a model and explain why existing models are not sufficient. Finally, we propose a new Zeno-free semantic model that allows mixing discrete and continuous behaviour in a rigorous way and provides for the compositional behavioural abstraction.

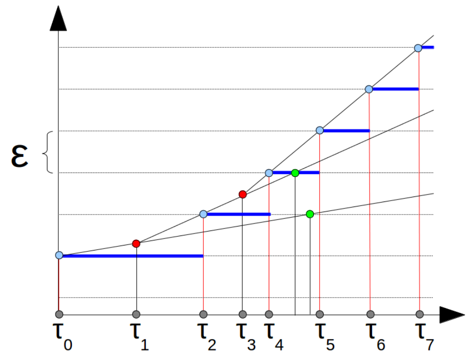


Figure 1: Semantics of a differential equation.

Although it is based on non-standard analysis, we explain how our semantic model can be used to develop hybrid system simulators.

Keywords: Hybrid Modeling Languages; Non-Standard Analysis; Models of Signals; Behavioral Abstraction; Operational Semantics

References

[1] Albert Benveniste, Timothy Bourke, Benoit Caillaud, and Marc Pouzet. Non-standard semantics of hybrid systems modelers. *Journal of Computer and System Sciences*, 78:877–910, May 2012. doi: 10.1016/j.jcss.2011.08.009.

[2] Simon Bliudze and Daniel Krob. Modelling of complex systems: Systems as dataflow machines. *Fundamenta Informaticae*, 91:1–24, 2009. doi: 10.3233/FI-2009-0001.

[3] Heinrich Rust. *Operational Semantics for Timed Systems: A Non-standard Approach to Uniform Modeling of Timed and Hybrid Systems*, volume 3456 of *Lecture Notes in Computer Science*. Springer, 2005. ISBN 3-540-25576-1. doi: 10.1007/978-3-540-32008-1.

[4] Bernard P. Zeigler and Jong Sik Lee. Theory of quantized systems: formal basis for DEVS/HLA distributed simulation environment. In *SPIE Proceedings*, volume 3369, pages 49–58, 1998.

An example of beneficial use of variable-structure modeling to enhance an existing rocket model

Alexandra Mehlhase Daniel Gomez Esperon Julien Bergmann Marcel Merkle
Technical University of Berlin, Department of Software Engineering
Ernst-Reuter-Platz 7, 10587 Berlin

This paper introduces a rocket model and discusses the advantages of refining it using a variable-structure approach to remodel computationally intensive parts.

The aim of our simulation is to predict the trajectory of a rocket, beginning with the ignition on earth's surface up until it reaches the moon as destination. The rocket is multi-staged, and as such consists of three booster modules and a payload module without means of propulsion. The model takes into account the chemical reactions in the boosters combustion chambers, which generate the thrust, the gravity of both earth and moon as well as atmospheric influences. First, we introduce a classical implementation of the model which is a rather stiff system of equations and does take quite a long time to simulate.

We are then introducing a variable-structure approach [1] which allows us to divide the rocket model into different modes. This is possible since the chemical reactions in the boosters only need to be regarded while starting a new stage. After a short time a steady state is reached and the chemical reactions, which also lead to the stiff system of equations, is not necessary anymore and they ca be taken out of the simulation. This leads to a faster simulation and less saved data volume without reducing the accuracy of the model.

Both versions of the model are implemented in Modelica and were simulated using Dymola as simulation environment. The DySMo framework [2], which supports the simulation of variable-structure models in common simulation environments, was used to facilitate the redesign.

The general benefits of the variable-structure approach are presented, and with the example of the rocket model we present that simulation time and the data volume of the simulation can be reduced while maintaining the accuracy of the simulation results.

References

[1] Breitenecker, F., Development of simulation software - from simple ode modelling to structural dynamic systems. In L. Louca, Y. Chrysanthou, Z. Oplatková, and K. Al-Begain, editors, Proceedings of the 22nd European Conference on Modelling and Simulation (ECMS 2008), pages 2037, 2008.

[2] Mehlhase, A., A Python framework to create and simulate models with variable structure in common simulation environments. In Mathematical and Computer Modelling of Dynamical Systems, December, 2013.

Efficient Monte Carlo simulation of stochastic hybrid systems

Marc Bouissou[1,5], Hilding Elmqvist[2], Martin Otter[3], Albert Benveniste[4]
[1]EDF R&D, 1 av. du Général de Gaulle, 92141 Clamart, France
[2]Dassault Systèmes AB, Ideon Science Park, SE-223 70 Lund, Sweden
[3]DLR, Institute of System Dynamics and Control, D-82234 Wessling, Germany,
[4]IRISA/INRIA, Campus de Beaulieu, 35042 Rennes Cédex, France
[5]Ecole Centrale Paris, Grande Voie des Vignes, 92295 Châtenay Malabry, France
Marc.Bouissou@edf.fr, Hilding.Elmqvist@3ds.com, Martin.Otter@dlr.de, Albert.Benveniste@inria.fr

This paper proposes an efficient approach to model stochastic hybrid systems and to implement Monte Carlo simulation for such models, thus allowing the calculation of various probabilistic indicators: reliability, availability, average production, life cycle cost etc. Stochastic hybrid systems can be considered, most of the time, as Piecewise Deterministic Markov Processes (PDMP). Although PDMP have been long ago formalized and studied from a theoretical point of view by Davis (*Davis 1993*), they are still difficult to use in real applications. The solution proposed here relies on a novel method to handle the case when the hazard rate of a transition λ depends on continuous variables of the system model, the use of an extension of Modelica 3.3 and on Monte Carlo simulation. We illustrate the approach with a simple example: a heating system subject to failures, for which we give the details of the modeling and some calculation results. We compare our ideas to other approaches reported in the literature.

Typical simulation results for the example are shown in the figure below. To perform a Monte Carlo simulation to estimate, for example, the mean temperature as a function of time, it is necessary to generate a large number of trajectories using different initial seeds for every simulation. This task can be performed in Dymola by using appropriate script functions (that are based on the algorithmic part of the Modelica language). A special Modelica/Dymola script has been implemented for this case to run the simulations and store the desired fractiles. In Figure 1, the mean value of the room temperature is shown, as well as the 1% and 99% fractiles at each time point respectively. 10 000 simulations were performed with 500 output points per simulation. On a notebook, these simulations took 25s.

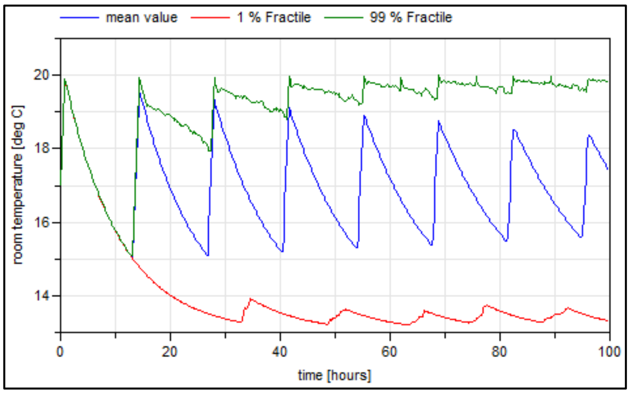


Figure 1: Statistics about room temperature (based on heating system subject to failures) obtained from 10000 trajectories

The Modelica BehaviorTrees Library

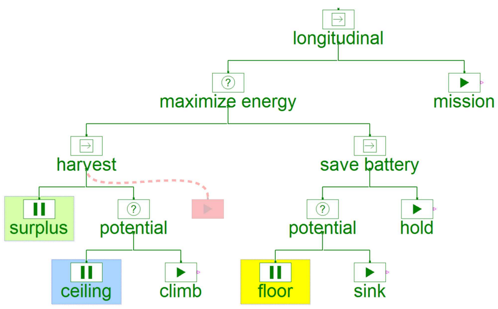
Mission Planning in Continuous-Time for Unmanned Aircraft

Andreas Klöckner
German Aerospace Center (DLR), Institute of System Dynamics and Control
82234 Weßling, Germany, Andreas.Kloeckner@dlr.de

Growing complexity of missions, environmental conditions, and UAV capabilities call for a flexible, scalable and intuitive scheme for UAS control systems and mission plans. Behavior trees have recently been proposed for this purpose. They are distinguished by their standardized structure providing a mission design scheme, which has been argued to combine important advantages of different schemes such as state machines and task planners.

However, conventional behavior tree implementations rely on *discrete-time* processing unsuitable for continuous-time simulation of long-term missions. In order to combine efficient long-term simulations with the capabilities of behavior tree mission plans, a *continuous-time* `BehaviorTrees` library was thus developed and implemented:

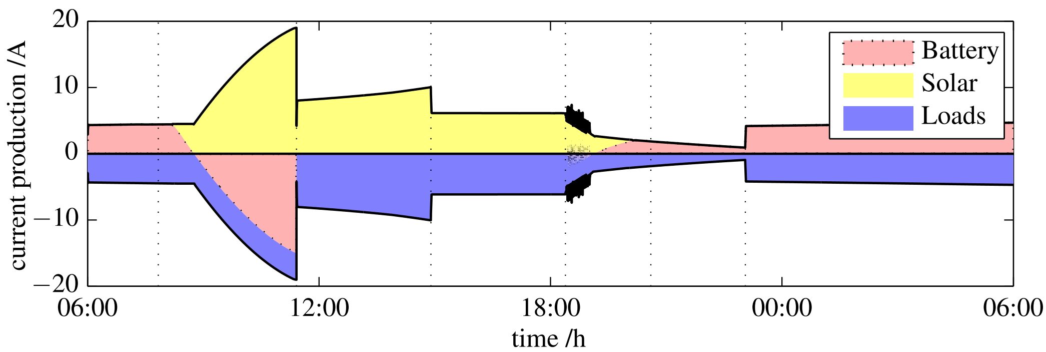
- It allows a simulator to chose large integration step-sizes as desired for long-term mission simulation. The formulation can be generalized to other languages supporting event notifications.
- A library of base tasks with clear internal and external interfaces allows the user to graphically design mission plans and also easily implement new task types.
- The 24 h-simulation with an integrated solar UAV model shown below underlines the modularity of the approach and its good performance.



(a) The example `BehaviorTrees` mission plan.

Configuration	CPU time	time- / state events	
Discrete	368 s	1442	249
Continuous	72 s	0	247
State graph	76 s	60	242
Direct inputs	66 s	25	183

(b) The continuous-time simulation is much faster than the discrete-time implementation. It is as fast as a reference state graph implementation and a direct simulation. Additionally, it does not require any time events.



(c) The simulation is run for 24h of flight with a 72 state continuous-time solar UAV model.

Multi-Level Library of Electrical Machines for Aerospace Applications

P. Giangrande C.I. Hill C. Gerada S.V. Bozhko
University of Nottingham
PEMC Group, Nottingham, NG7 2RD, UK.
p.giangrande@nottingham.ac.uk c.hill@nottingham.ac.uk

An electrical machine library, developed within the framework of the European project *Actuation 2015*, is presented in this paper. The library has been developed adopting a multi-level approach, in order to minimize the models complexity and reduce the computational time. Multi-level approach consists in creating several models of the same electrical machine topology, with different levels of complexity. Indeed, model complexity increases at higher modelling levels and each model takes into account specific physical effects. In addition to the fundamental behavior, the presented models address physical effects such as losses, thermal behavior, magnetic saturation, torque ripple and fault conditions (i.e. short-circuits, open-circuits and permanent magnet demagnetization).

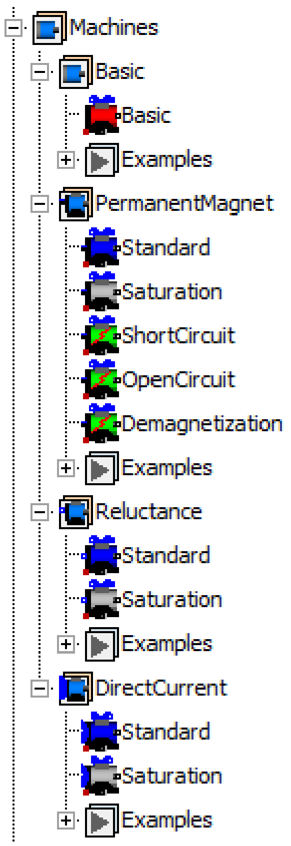


Figure 1: Structure of the `Electrical Machines` library as part of the `Actuator` library.

Considering that several models of the same machine topology are available within the library, the interchangeability among the model levels is crucial, in order to investigate different physical effects by simply replacing the model level. For this reason, the interchangeability has been ensured by using common interfaces among the different model levels. Moreover, all the developed models are power balanced and can work indifferently representing motor or generator.

The `Electrical Machines` library has been implemented using Modelica as modelling language. As reported in figure 1, the library is organized as three packages (one for each machine topology) plus the package `Basic`, which considers a generic alternating current electrical machine. The considered machine topologies are permanent magnet synchronous machine (package `PermanentMagnet`), synchronous reluctance machine (package `Reluctance`) and direct current machine (package `DirectCurrent`). Each package contains the sub-package `Examples`, which provides several study cases, helpful for highlighting the features of each modelling level.

The presented work is mainly focused on describing the package for permanent magnet synchronous machine because they are characterized by an excellent efficiency, together with a high power density. These features make permanent magnet synchronous machines very attractive for aerospace applications.

Physical effects included in the models are discussed and their implementation is detailed. In order to highlight the implemented physical effects and confirm the models effectiveness, Dymola simulation results are provided for several operating conditions.

Modelica for large scale aircraft electrical network V&V

Martin R. Kuhn Yang Ji
German Aerospace Center (DLR), Institute of System Dynamics and Control
Münchnerstr. 20, 82234 Weßling
martin.kuhn@dlr.de yang.ji@dlr.de

The model based design approach is a key factor for more efficient aircraft design with its growing demand to optimize the complex physical systems containing mechanical, electrical, hydraulic, thermal, control, electric power or process-oriented sub components. Especially the more electric aircraft concept relies on incorporating high quality system models in the complete aircraft design process. The process itself briefly can be divided into 4 major phases: concept phase, system specification phase, system development phase and system verification phase. The model types and level of detail change for every phase. The aircraft electrical network validation and verification process strongly relies on software for detailed and numerical complex modeling, simulation and analysis of network components and systems. Substantial efforts were made to reach platform independence and link simulation tools each with special strengths and dedicated for specific domains. Especially the FMI standard was a major step forward and was verified to improve an aircraft systems design process. Nevertheless, for the sake of performance and transparency, industrial processes often rely on a single common tool.

The software used in an aircraft project for the systems integration validation and verification (V&V) process is defined by the airframer for all model suppliers and contributors. While Modelica has found attraction in the automotive sector, it is not the standard for detailed simulation in aeronautic industry yet. Inspired by the success in the prior design phases, a study was performed in the context of the CleanSky project to evaluate the potential and performance of Modelica and the commercial tool Dymola for electrical V&V . In this paper we give an overview of the necessities of the infrastructure which had to be developed. We demonstrate the modeling and the simulation results of component stand-alone tests as well as the tests of an integrated aircraft power network. Necessary tools are addressed and lessons learned from the study are documented. It is the aim of this paper to rise awareness of the needs to conduct V&V studies.

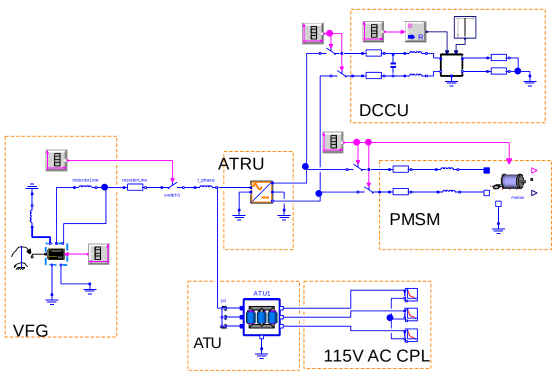


Figure 1: An integrated electric power network for MEA

Implementation of a Modelica Library for Simulation of Electromechanical Actuators for Aircraft and Helicopters

Franciscus L.J. van der Linden
German Aerospace Center (DLR),
Institute of System Dynamics and
Control, Münchner Straße 20
82234 Weßling, Germany
Franciscus.vanderlinden@dlr.de

Clemens Schlegel
Schlegel Simulation GmbH
Meichelbeckstr. 8b
85356 Freising, Germany
cs@schlegel-simulation.de

Markus Christmann
EADS Innovation Works, TCC6
81663 Munich, Germany
markus.christmann@eads.net

Gergely Regula
MTA SZTAKI
13-17 Kende St.
1111 Budapest, Hungary
gergely.regula @sztaki.mta.hu

Chris I. Hill
University of Nottingham
Aerospace Technology Centre,
Innovation Park
Nottingham. NG7 2TU. UK
c.hill@nottingham.ac.uk

Paolo Giangrande
University of Nottingham
Tower Building,
University Park
Nottingham. NG7 2RD. UK
p.giangrande@nottingham.ac.uk

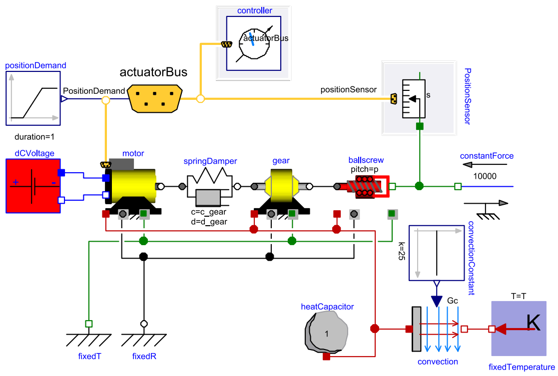
Jean-Charles Maré
Institut National des Sciences Appliquées
135 Avenue de Rangueil
31077 Toulouse Cedex 4, France
jean-charles.mare@insa-toulouse.fr

Imanol Egaña
IK4 TEKNIKER
Calle Iñaki Goenaga, 5
20600 Eibar, Gipuzkoa, Spain
imanol.egana@tekniker.es

Protecting the environment and providing efficient onboard energy supply is amongst the top targets in the development of future aircraft. Various research initiatives have been launched in recent years to get closer to this goal. The ACTUATION2015 project [1] will complete this approach by focusing on Electro Mechanical Actuator (EMA) technologies. EMAs are mandatory in order to eliminate hydraulic circuits, pumps and reservoirs. The goal of the A2015 library presented in this paper is to develop a Modelica based, tool-independent standard for electromechanical actuator libraries.

Since the intended use of the library includes all development stages from concept assessment to virtual design validation several models of different scope and level of detail are implemented for the core EMA components (multi-level approach). Five modeling levels are predefined, mostly associated with nonlinearities included and events triggered. The library does allow for modeling redundant actuators (which are needed because EMAs have lower reliability than conventional hydraulic actuators) and uses a unified approach for component failure injection.

The A2015 library contains models from various domains: Electrical (inverters, motors), mechanical (rotation to rotation and rotation to translation transformers), sensors (position, speed, force, etc.), thermal (heat sinks, housings), and control (e.g. force fight compensator). Selected library components and their implementation are described in the paper. A typical application example is given and discussed (A320 aileron).



References

[1] www.actuation2015.eu

An Optimization Framework for Dynamic Hybrid Energy Systems

Wenbo Du[1] Humberto E. Garcia[1] Christiaan J.J. Paredis[2]

wenbo.du@inl.gov humberto.garcia@inl.gov chris.paredis@gatech.edu

[1]Idaho National Laboratory
750 University Boulevard, Idaho Falls, ID 83415-3570, USA

[2]Georgia Institute of Technology
813 Ferst Drive, Atlanta, GA 30332-0405, USA

A computational framework for the efficient analysis and optimization of dynamic hybrid energy systems (HES) is developed. A microgrid energy system with multiple inputs and multiple outputs (MIMO) is modeled using the Modelica language in the Dymola environment. The optimization loop is implemented in MATLAB, with the FMI Toolbox serving as the interface between the computational platforms. Two characteristic optimization problems are selected to demonstrate the methodology and gain insight into the system performance. The first is an unconstrained optimization problem that optimizes intrinsic properties of the base generation, power cycle, and electrical storage components to minimize variability in the HES. The second problem takes operating and capital costs into consideration by imposing linear and nonlinear constraints on the design variables. Variability in electrical power applied to high temperature steam electrolysis is shown to be reduced by 18% in the unconstrained case and 11% in the constrained case. The preliminary optimization results obtained in this study provide an essential step towards the development of a comprehensive framework for designing HES.

Industrial application of optimization with Modelica and Optimica using intelligent Python scripting

K. Dietl[a], S. Gallardo Yances[a], A. Johnsson[b]
J. Åkesson[c], K. Link[a], S. Velut[c]
[a]Siemens AG, Energy Sector, Erlangen, Germany
[b]Lund University, Department of Automatic Control, Lund, Sweden.
[c]Modelon AB, Lund, Sweden.

This paper shows how different kinds of optimization related task such as offline optimization or optimal control are solved using a combination of Modelica, Optimica, JModelica.org [1] and Python [2]. The application examples presented in this paper are all real industrial applications in the field of Combined Cycle Power Plants, namely the water-steam cycle.

Figure 1 shows a part of a water-steam cycle used in an offline optimization of a plant start-up.

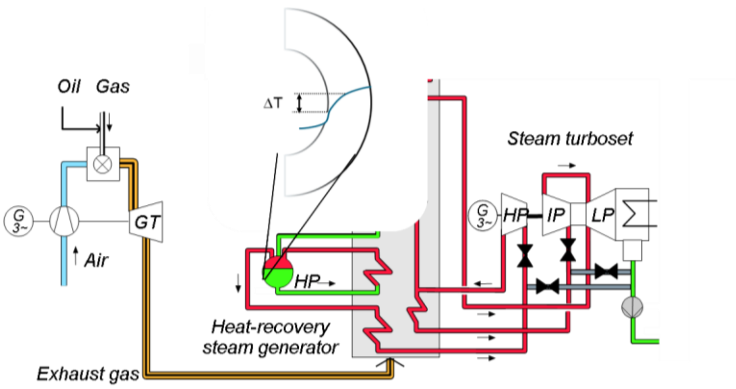


Figure 1: Model for startup optimization

Besides the offline optimization of a power plant startup, the paper also includes an example of a nonlinear model predictive control with state estimation using the extended Kalman filter and a parameter estimation of a Modelica model representing the plant layout shown in Figure 1.

The focus of the paper lies on the interaction of Modelica, Optimica, JModelica and Python and demonstrates the suitability of Python as scripting environment in this context.

References

[1] JModelica.org, http://jmodelica.org/, viewed 2013-12-05.

[2] Python Software Foundation. Python Pro-gramming Language - Official Website, http://www.python.org/, 2012, viewed 2013-12-05

Simulation of Smart–Grid Models using Quantization–Based Integration Methods

X. Floros[a1] F. Bergero[b1] N. Ceriani[c] F. Casella[c] E. Kofman[b] F. E. Cellier[a]

[a]Department of Computer Science, ETH Zurich, Switzerland
[b]CIFASIS-CONICET, Rosario, Argentina
[c]Politecnico di Milano, Dipartimento di Elettronica, Informazione e Bioingegneria, Italy

Concepts such as **smart grids, distributed generation and micro–generation of energy** are becoming increasingly important as emerging trends in the design, management and control of energy systems. Appropriate modeling and design, efficient management and control strategies of such systems are currently being studied. In this line of research a very important enabling component is efficient and reliable simulation.

However those **energy models are typically large, stiff and exhibiting heavy discontinuities**, and at the same time consist of interconnected multi–domain subsystems encompassing electrical, thermal, and thermo-fluid models. Object-Oriented (O–O) languages such as Modelica are obviously well-suited for the modeling of such systems; **however, traditional state-of-the-art hybrid differential algebraic equation solvers cannot efficiently simulate these systems** especially when their size grows to the order of hundreds, thousands, or even more interconnected units.

The goal of this paper is to show, through a couple of exemplary case studies, that **Quantized State System (QSS) integration methods are ideally suited to solve models of smart grid systems**, as they scale up better than traditional methods with the system size, and provide time savings of several orders of magnitude, while achieving comparable numerical precision.

More specifically, **in both cases QSS methods outperform DASSL (and Runge-Kutta) by more than two orders of magnitude in terms of simulation speed**, while at the same time, achieving a comparable simulation error, as seen in Fig. 1. Furthermore, the **QSS methods scale linearly with system size, while DASSL scales quadratically**.

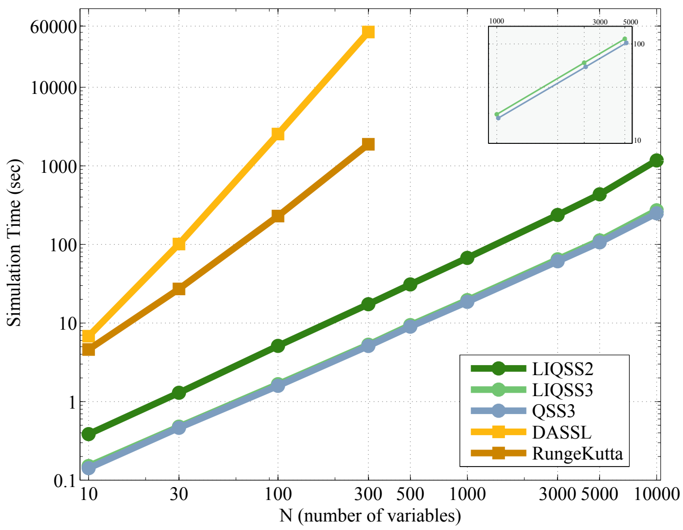


Figure 1: Simulation time for varying size of the Energy Market model. All algorithms achieve a mean error on the order of 10^{-5}.

On the Simulation of Offshore Oil Facilities at the System Level

Joris Costes[1,2] Jean-Michel Ghidaglia[2] Philippe Muguerra[3] Keld Lund Nielsen[4]
Xavier Riou[3] Jean-Philippe Saut[1] Nicolas Vayatis[2]

[1]Eurobios SCB, 86 avenue Lénine, 94250 Gentilly, France
{joris.costes, jpsaut}@eurobios.com

[2]CMLA, ENS Cachan, UMR 8536 CNRS, 61 avenue du président Wilson, 94235 Cachan cedex, France
{jmg, vayatis}@cmla.ens-cachan.fr

[3]eni saipem, 7 avenue de San Fernando, 78180 Montigny-le-Bretonneux, France
{Philippe.MUGUERRA,Xavier.RIOU}@saipem.com

[4]eni spa, exploration & production division, San Donato Milanese (Mi), Italy
Keld.Lund.Nielsen@eni.com

Offshore oil facilities are complex systems involving elaborate physics combined with stochastic aspects related, for instance, to failure risk or price variation. Although there exist many dedicated software tools to simulate flows typically encountered in oil exploitations, there is still no tool that combines physical (mostly engineering fluid mechanics) and risk simulation. Such a tool could be useful to engineers or decision makers for specification, design and study of offshore oil facilities. We present a first step towards the creation of such a tool. Our current simulator is based on new Modelica components to simulate fluid flows and on stochastic simulation at a higher level, for modeling risk and costs. Modelica components implement physical models for single and two-phase flows in some typical devices of an offshore field. The risk simulation is performed with Scilab and receives data from the Modelica simulation. It uses Markov chains and statistical indicators to assess performance and resilience of the system over several months or years of operation. The presented simulation framework can be used to compare different designs of an offshore oil facility, in order to choose the most productive and/or robust, depending on the choice of the performance evaluator.

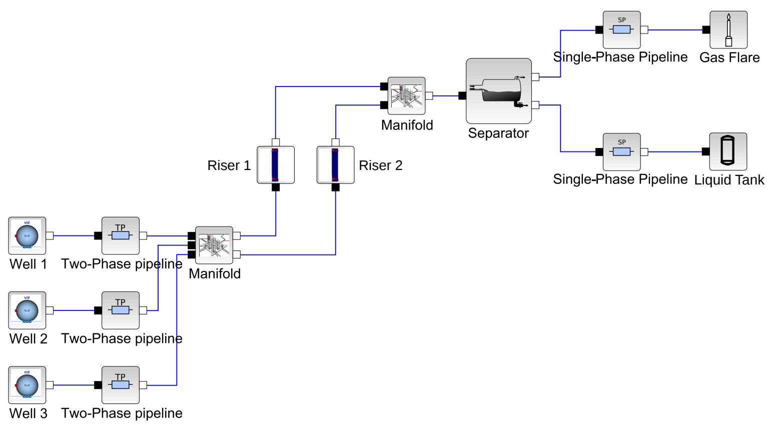


Figure 1: An example of simulated system, which includes some typical components of an offshore facility (lines, risers, separator, etc.)

Parameter Selection in a Combined Cycle Power Plant

Niklas Andersson[a] Johan Åkesson[b,c] Kilian Link[d]
Stephanie Gallardo Yances[d] Karin Dietl[d] Bernt Nilsson[a]
[a]Lund University, Department of Chemical Engineering, Lund, Sweden
[b]Lund University, Department of Automatic Control, Lund, Sweden
[c]Modelon AB, Lund, Sweden
[d]Siemens AG, Erlangen, Germany

A combined cycle power plant are modeled and considered for calibration. The dynamic model, aimed for start-up optimization, contains 64 candidate parameters for calibration. The number of parameter sets that can be created are huge and an algorithm called subset selection algorithm[2] is used to reduce the number of parameter sets. The algorithm investigates the numerical properties of a calibration from a parameter Jacobean estimated from a simulation of the model with reasonably chosen parameter values. The calibrations were performed with a Levenberg-Marquardt algorithm considering the least squares of eight output signals. The parameter value with the best objective function value resulted in simulations in good compliance to the process dynamics. The subset selection algorithm effectively shows which parameters that are important and which parameters that can be left out. A combined cycle power plant are modeled and considered for calibration. The dynamic model, aimed for start-up optimization, contains 64 candidate parameters for calibration. The number of parameter sets that can be created are huge and an algorithm called subset selection algorithm is used to reduce the number of parameter sets. The algorithm investigates the numerical properties of a calibration from a parameter Jacobean estimated from a simulation of the model with reasonably chosen parameter values. The calibrations were performed with a Levenberg-Marquardt algorithm considering the least squares of eight output signals. The parameter value with the best objective function value resulted in simulations in good compliance to the process dynamics. The subset selection algorithm effectively shows which parameters that are important and which parameters that can be left out.

References

[1] Åkesson, J., and Årzén, K.-E., Gäfvert, M., Bergdahl, T., Tummescheit, H., nov 2010. Modeling and Optimization with Optimica and JModelica.org—Languages and Tools for solving large-scale dynamic optimization problem. Computers and Chemical Engineering 34 (11), 1737–1749.

[2] Cintrón-Arias, A., Banks, H. T., Capaldi, A., Lloyd, A. L., 2009. A Sensitivity Matrix Based Methodology for Inverse Problem Formulation. Journal of Inverse and Ill-Posed Problems, 17(6), 545-564.

Restarting algorithms for simulation problems with discontinuities

Fatemeh Mohammadi Carmen Arévalo Claus Führer*
Numerical Analysis, Center of Mathematical Sciences, Lund University
Sölvegatan 18, SE-22100 Lund, Sweden

Abstract

Modelica's language support includes so-called events for describing discontinuities. Modern integrating environments, like Assimulo, provide elaborate event detection and event handling methods. In addition, the overall performance of a simulation of models with discontinuities (hybrid models) depends strongly on the methods for restarting the integration after an event detection. The present paper reviews two restarting methods for multistep methods, both based on Runge–Kutta starters, and presents preliminary first experiments with Assimulo and LSODAR as a proof of concept, which motivates to apply the technique to hybrid systems described in Modelica and simulated by JModelica.org/PyFMI and Assimulo [1, 3, 2].

Keywords: events, discontinuities, hybrid systems, multistep method, Runge–Kutta method, simulation restart

References

[1] Johan Åkesson, Magnus Gäfvert, and Hubertus Tummescheit. JModelica—an open source platform for optimization of modelica models. In *Proceedings of MATHMOD 2009 - 6th Vienna International Conference on Mathematical Modelling*, Vienna, Austria, February 2009. TU Wien.

[2] Christian Andersson. Assimulo: a new Python based class for simulation of complex hybrid DAEs and its integration in JModelica.org. Master's thesis, Lund University, 2011.

[3] Christian Andersson, Johan Åkesson, Claus Führer, and Magnus Gäfvert. Import and export of functional mock-up units in JModelica.org. In *8th International Modelica Conference 2011*, Dresden, Germany, March 2011.

*partly supported by LCCC - Lund Center for Control of Complex Engineering Systems

Discontinuities handled with events in Assimulo, a practical approach

Emil Fredriksson* Christian Andersson*,** Johan Åkesson*,***
*Modelon AB, Sweden
Lund University, Sweden
**Department of Numerical Analysis
***Department of Automatic Control

Often integrating ordinary differential equations or differential algebraic equations (DAE) do not constitute the problem alone. A common complement is finding the root of an algebraic function (an event function) that depends on the states of the problem. This formulation of a model enables the possibility of including discontinuities, an important part of the Functional Mock-up Interface (FMI) standard which allows hybrid models of differential algebraic equations. The problem of root-finding during integration is however difficult, both in a theoretical aspect and as a software problem.

The Illinois algorithm was chosen as a basis for the event algorithm. An important improvement is to apply the domain formulation, meaning that instead of defining an event as a change in sign (zero-crossing formulation) for the event function, g, it is defined as a change in domain from $g > 0$ to $g \leq 0$ or vice versa, this is consistent with the FMI standard. An advantage is that the zero is no longer a special case and also, more importantly, events caused by event functions becoming exactly zero for a finite time is found correctly.

An implementation of software for the event algorithm is done in Assimulo, a Python/Cython wrapper for integrators. This enabels event location for numerous integrators and therefore also support for simulating FMUs by using PyFMI.

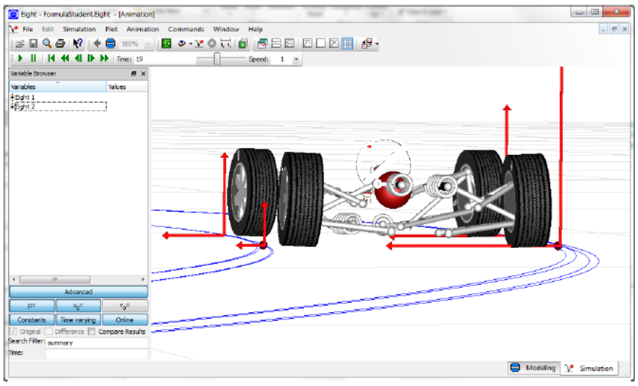


Figure 1: A picture of the car in Dymola, used for the benchmark.

These integrators are benchmarked with industrial relevant models to verify that they work well together with the event algorithm and that they find events accurately, especially for the case where the event functions becomes zero for a finite time. It is also important that the performance does not deteriorate with the modifications to the event algorithm. The models used in the benchmark are a clutch with inputs, a Furuta pendulum and a racing car.

Comparing Sundials using its zero-crossing event algorithm and Sundials using the articles event algorithm no significant loss in performance was seen. Additionally, the events of the clutch model with inputs was found significantly better with the articles event algorithm. The other integrators benchmarked also performed well.

Noise Generation for Continuous System Simulation

Andreas Klöckner   Franciscus L. J. van der Linden   Dirk Zimmer
{andreas.kloeckner, franciscus.vanderlinden, dirk.zimmer}@dlr.de

German Aerospace Center (DLR), Institute of System Dynamics and Control,
82234 Weßling, Germany

The problem of introducing *disturbances to a nominal system* eventually becomes an issue, when simulating real-world systems. Especially, when dealing with controlled systems, important tasks are to check whether the controller is able to reject realistic disturbances, and to assess the performance of the system including noise. The problem is not limited to the field of control design, but is also of interest in e.g. specification of aircraft airworthiness requirements with respect to turbulence, estimation of the power outcome of wind energy farms or when interpreting contaminated sensor readings of experiments.

This noise contribution must be *specified carefully*, because it can have a strong impact on the system's performance. However, there are no convenient means of specifying noise properties in Modelica, such that typical approaches implement ad-hoc modifications of a simple random signal.

The *simulation performance* of the model should not be decreased by adding noise to the system. However, injecting noise into a continuous system typically decreases the simulation speed drastically, because standard noise generators are sampled systems, which generate time events by definition. These time events lead in most cases to integrator restarts, imposing a big penalty on simulation performance.

In this work, we present ways to solve the two issues outlined above in an integrated library:

1. We describe a general procedure to *specify a suitable noise signal* by means of selecting a high-quality random number generator, a probability distribution and a power spectrum.

2. By providing a *continuous noise* signal formulation using a sample-free generator and a smooth interpolation, we provide means for continuous noise generation. Avoiding events and using a smoothly filtered signal speeds up the simulation as compared to standard methods.

3. The methods and processes are integrated in a *library with convenient user interfaces*. This enables a user to easily specify a desired noise signal and to use it in complex simulation models.

Figure 1 shows the timing of an actuator simulation without noise, with conventional `LinearSystems` noise and with the new `Noise` library. The noisy simulation is relieved of unnecessary time events, using the `Noise` library. Its speed is increased by a factor of 2.5.

		Events	
Model	Time	Time	State
No Noise	0.037 s	1	12
`LinearSystems`	27.5 s	15000	1642
`Noise`	10.5 s	1	1791

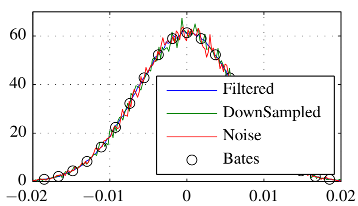


Figure 1: Timing of an actuator simulation and the distribution of a random variable

A physical solution for solving the zero-flow singularity in static thermal-hydraulics mixing models

Daniel Bouskela   Baligh El Hefni
EDF R&D
6, quai Watier F-78401 Chatou Cedex, France
daniel.bouskela@edf.fr baligh.el-hefni@edf.fr

For the 0D-1D modelling of thermal-hydraulics systems, it is common practice to use static mixing models to compute the mixing specific enthalpy in fluid junctions such as mergers or splitters. However, this simplification leads to a well known singularity when the mass flow rate inside the junction goes to zero. The origin of the singularity is explained, and a rigorous physical solution is proposed to eliminate the singularity. A prototype implementation has been developed in the ThermoSysPro library for power plant modelling that illustrates the interest of the proposed solution, shows the impact on the structure of the library and enables to evaluate the computing overhead with respect to several possible variants.

Keywords: *thermal-hydraulics; mixing models; convection; diffusion; ThermoSysPro*

Advanced Hybrid Model for Borefield Heat Exchanger Performance Evaluation, an Implementation in Modelica

Damien Picard[1] Lieve Helsen[1]
[1]KU Leuven, Department of Mechanical Engineering,
Division of Applied Mechanics and Energy Conversion
Celestijnenlaan 300A, box 2421, BE-3001 Heverlee, Belgium

Accurate and computationally efficient borefield models are important components in building energy simulation programs. They have not been implemented in Modelica so far. This paper describes the implementation of an innovative approach to model borefields with arbitrary configuration having both short-term (minutes) and long-term accuracy (decades) into Modelica. The model allows coaxial, single and double U-tube borehole types.

A step response is calculated using a combination of a short-term response model which takes into account the transient heat transfer in the heat carrier fluid, the grout and the immediately surrounding ground, and a long-term response model which calculates the boreholes interactions. Moreover, an aggregation method is implemented to speed up the calculations and to make the simulation time independent of the number of boreholes. Thanks to its aggregation method, the implemented model is about twelve times faster than the borehole model of the Buildings library [1] for the case of a single borehole and about 60 times faster for the case of three boreholes in series (see Fig. 1).

Validation shows good results and very high computational efficiency.

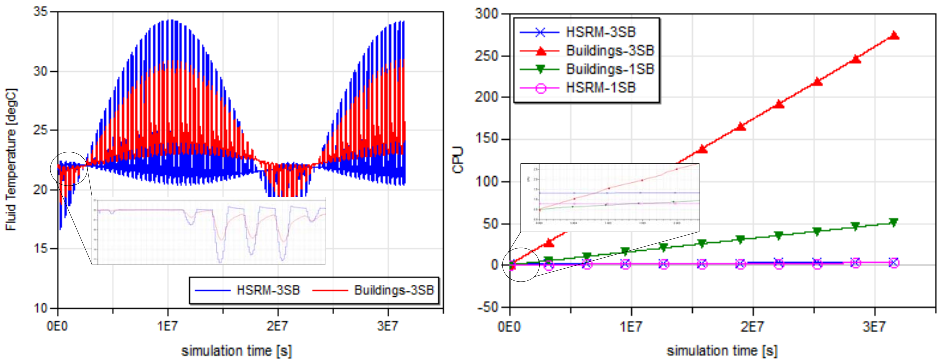


Figure 1: Left: CPU comparison between the new model (*HSRM*) and the model form the Buildings library (*Buildings*) for a single borehole (*1SB*) and for three boreholes in serie (*3BH*). Right: heat carrier temperature for *HSRM-3BH* and *Buildings-3BH*.

References

[1] M. Wetter, W. Zuo, T. Nouidui, and X. Pang. Modelica buildings library. *Journal of Building Performance Simulation*, 0:1–18, 2013.

Superheat Control with a Dynamic Inverse Model

Andreas Varchmin[1] Manuel Gräber[2] Wilhelm Tegethoff[1,2] Jürgen Köhler[1]
[1] Technische Universität Braunschweig, Institut für Thermodynamik
[2] TLK-Thermo GmbH
a.varchmin@tu-braunschweig.de

Superheat control has influence on the coefficient of performance (COP), the stability and the compressor endurance of a vapor compression cycle. It is a challenging task since the system dynamics of a refrigeration cycle are highly nonlinear and the control targets can be contrary. In an increasing number of applications electronic expansion valves (EXV) are used. With EXVs superheat and thus stability and COP can be controlled directly in contrast to thermostatic expansion valves (TXV) or orifice valves. New control approaches can benefit from this additional degree of freedom.

It raises the question if simulation models can be used for feedforward control to fulfill this function. For building a feedforward control structure a simulation model needs to be inverted. In this paper a dynamic, continuous submodel of a refrigeration cycle, consisting of models for expansion valve and evaporator, is inverted. The resulting controller is tested in a model-in-the-loop environment and applied on an automotive refrigeration cycle. The advantage of a dynamic inverse model in contrast to a static one is pointed out. Also the results are compared to a standard PI controller.

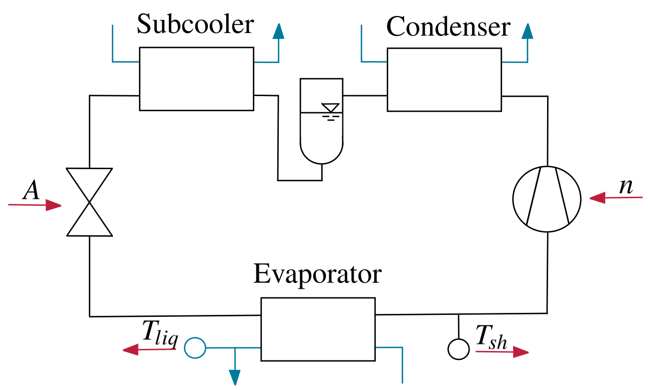


Figure 1: Modelled Refrigeration Cycle

Adsorption energy systems library - Modeling adsorption based chillers, heat pumps, thermal storages and desiccant systems

Uwe Bau[1] Franz Lanzerath[1] Manuel Gräber[2] Stefan Graf[1]
Heike Schreiber[1] Niklas Thielen[1] André Bardow[1]

[1]RWTH Aachen University, Institute of Technical Thermodynamics
Schinkelstr. 8, 52062 Aachen, Germany

[2] TU Braunschweig, Institute of Thermodynamics
Hans-Sommer-Straße 5, 38106 Braunschweig, Germany

andre.bardow@ltt.rwth-aachen.de

A library for dynamic modeling adsorption based thermal systems like chillers, heat pumps, thermal storages or desiccant units is presented. Adsorption devices can serve a wide range of applications but usually consist of the same basic components. By modeling these basic components, the presented model library allows to investigate any interesting topology. Thereby this adsorption library gives the user the opportunity to design and optimize adsorption systems quickly and efficiently. To demonstrate the flexibility of the library and the accuracy of the simulations, three validated examples are presented: A desiccant unit; a thermal storage; and an adsorption chiller.

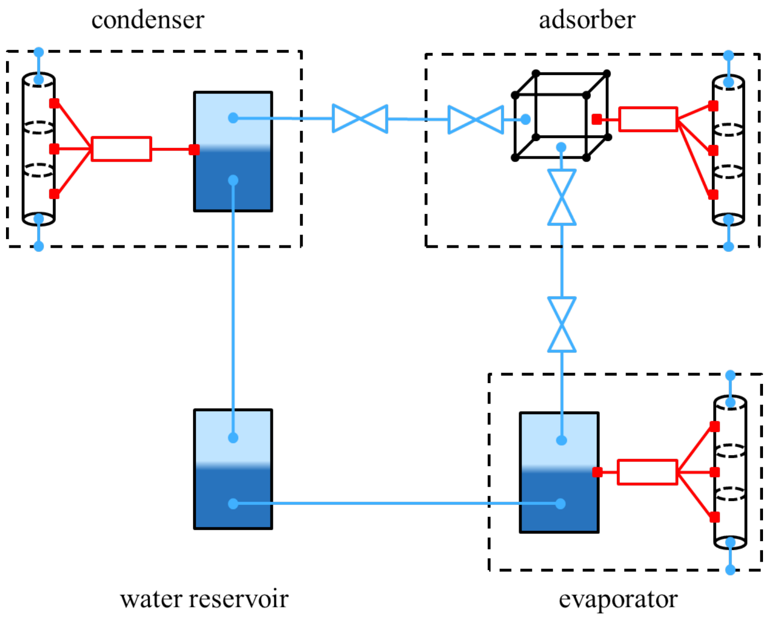


Figure 1: Scheme of adsorption chiller consisting of an adsorbent, evaporator, condenser, mass resistances, heat resistances, and a water reservoir model

A new Implementation of the N-D Lookup Tables

Torsten Sommer   Markus Andres   Stephan Diehl
Modelon GmbH
Agnes-Pockels-Bogen 1
D-80992 Munich, Germany
torsten.sommer@modelon.com   markus.andres@modelon.com   stephan.diehl@modelon.com

The HDF5Table library is an open-source solution for the efficient handling, exchange and interpolating access of typical data sets in system simulation. The library consists of C-functions, python scripts and examples and can be used with different applications like Modelica or Simulink. Furthermore a comprehensive set of tools that allows the user to create, migrate, edit, compare and manage the datasets is available. The application range covers data import from measurements or other simulations, integration of datasets in preprocessing routines, the usage of the datasets in the simulation and the post processing of simulation results. To eliminate a major source of errors after data exchange between simulation tools or different companies and to validate the datasets each dataset can have a physical unit and quantity attached to it. The table data can be easily accessed with different methods for inter- and extrapolation. To persist and exchange the data sets a subset of the HDF5 [1] standard is used. With the HDF5 API the data access is fast for large files with many variables containing millions of values and the datasets can be opened in many other tools.

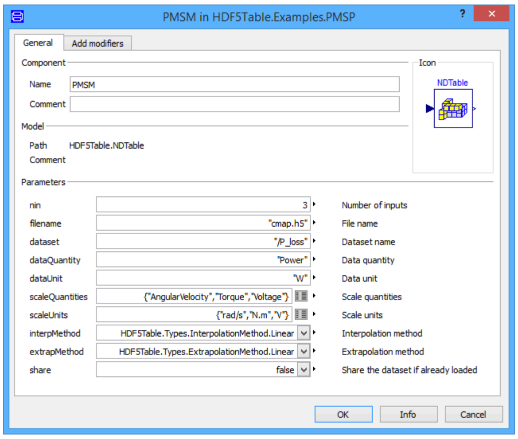


Figure 1 NDTable block parameters

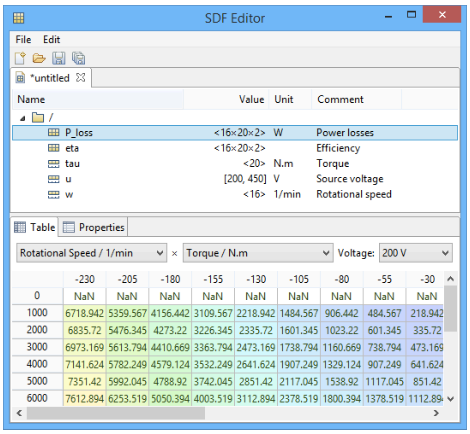


Figure 2 HDF5 dataset editor

A number of solutions exist for Modelica [2] and other simulation platforms that suffer from different limitations and problems the proposed implementation together with a set of supporting tools is trying to solve. The above figures show the parameters dialog of the Modelica block and the corresponding HDF5 data file that contains the three-dimensional table with scales, units and quantities.

References

[1] HDF5 Software Documentation, http://www.hdfgroup.org/HDF5/doc/index.html

[2] Call for Quotation of an Open Source Implementation of the MSL Table Interpolation Blocks, https://www.modelica.org/news_items/call-texts-to-improve-modelica-2012/2012-12-20-Call-for-quotation-for-MSL-tables.pdf/at_download/file

Remarks on the Implementation of the Modelica Standard Tables

Thomas Beutlich Gerd Kurzbach Uwe Schnabel
ITI GmbH
Schweriner Straße 1, 01067 Dresden, Germany
{beutlich, kurzbach, schnabel}@itisim.com

This article reveals some implementation details regarding the C code of the revised table interpolation blocks released with the Modelica Standard Library (MSL) 3.2.1. This new table implementation was named *Modelica Standard Tables* and comprises the following four blocks for univariate and bivariate interpolation

- Modelica.Blocks.Sources.CombiTimeTable,
- Modelica.Blocks.Tables.CombiTable1D,
- Modelica.Blocks.Tables.CombiTable1Ds and
- Modelica.Blocks.Tables.CombiTable2D.

The emphasis is placed on the unique features of the CombiTimeTable which are the discontinuities by time events and the periodic extrapolation (Fig. 1). For instance, periodic and discontinuous signals like saw-tooth or square-wave w.r.t. simulation time can be modeled in a very convenient and compact way. However, the numerically stable detection of periodic time events is rather tricky since floating-point arithmetic must not be used to detect them.

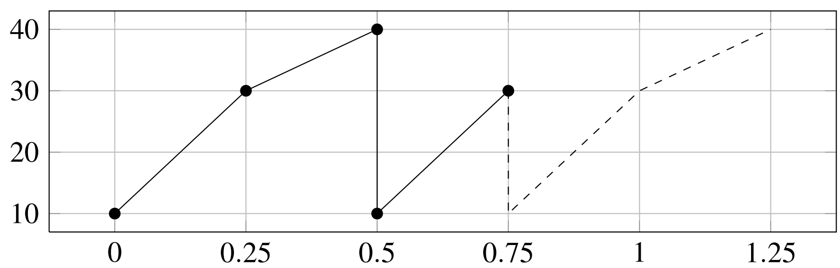


Figure 1: There are two time events per period in case of linear interpolation and periodic extrapolation of the 5×2 time table [0, 10; 0.25, 30; 0.5, 40; 0.5, 10; 0.75, 30].

Besides the CombiTimeTable basic information on the univariate and bivariate interpolation by Akima splines [1, 2] and the available table array memory optimization options are summarized. Last but not least, the remaining newly implemented table interpolation features are also mentioned.

References

[1] Hiroshi Akima. A new method of interpolation and smooth curve fitting based on local procedures. *J. ACM*, 17(4):589–602, October 1970.

[2] Hiroshi Akima. A method of bivariate interpolation and smooth surface fitting based on local procedures. *Commun. ACM*, 17(1):18–20, January 1974.

The DLR Visualization Library

Recent development and applications

Matthias Hellerer   Tobias Bellmann   Florian Schlegel
German Aerospace Center (DLR), Institute of System Dynamics and Control
82234 Wessling, Germany

The visualization of simulation data is an important element of advanced simulations. With the increasing complexity of modern multi-body simulations, concerning for example flexible bodies, thermal dissipation, or contacts, the demands for a realistic, real-time capable visualization of the simulation also rise. Using Modelica as modeling language allows the user to pack model functionality in reusable sub-models, e.g. as a replaceable block for a suspension, an engine, etc. This also enables the integration of visualization definitions into the single sub-models, eliminating the need for an additional visualization definition in a separate program.

In 2009, the "DLR Visualization Library" for Modelica has been introduced under the now deprecated name "DLR External Devices Library". The library features a variety of visualization blocks to be used directly as visualizers within Modelica. Visualization elements like configurable rigid bodies (e.g. sphere, box, gearwheel) or rigid and flexible bodies generated from CAD files can be displayed by connecting their respective Visualizer block to the corresponding frame in the multi-body model. Using the C-interface of Modelica, the visualization information is transmitted to the external visualization software which is started automatically with the simulation and allows the user to replay and observe the simulation run in real-time and export it as video.

New additions include among others: dynamic textures, rendering camera images or videos on flexible surfaces, Head-Up-Displays for presenting information and registering user inputs, a collision detection system, returning information about the 3D environment back to the simulation, a particle system, displaying non-rigid elements such as fire and integration of the Oculus Rift virtual reality device along side derived systems such as a wheel ground contact model and a flexible, interactive trajectory planner.

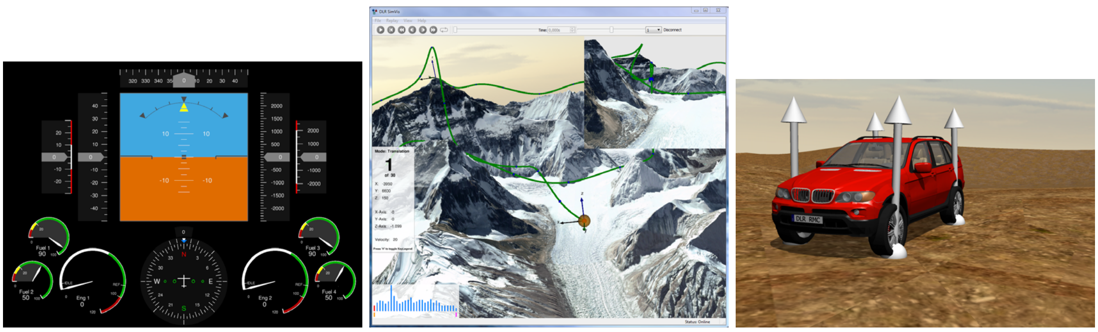


Figure 1: Examples of advanced visualizations; left to right: a complex pilots HUD with attitude indicator in the center, an flexible, interactive trajectory planner showing a path above Mt. Everest and a car driving on a 3D shape using collision detection and a contact model

Automated Modelica Package Generation of Parameterized Multibody Systems in CATIA

Daniel Baumgartner, Andreas Pfeiffer
German Aerospace Center (DLR), Institute of System Dynamics and Control
82234 Wessling, Germany
Daniel.Baumgartner@dlr.de, Andreas.Pfeiffer@dlr.de

In early stages of the product development process computer-aided design (CAD) and multibody simulation (MBS) work concurrently to build a virtual mechanical system. While CAD handles the geometric design and space analysis, MBS leads to a deeper understanding of the dynamic behavior of the future system. The CAD system has to provide physical and geometrical data, such as mass, inertia and connecting frames in order to improve simulation results. Automation at this point helps to create consistent simulation and design models and shortens the amount of time needed to produce realistic simulation results. Based on Visual Basic for Applications (VBA) a method is implemented to automatically generate an isolated Modelica model package or a single Modelica model from CATIA assemblies or parts. The introduction of design controlling parameter variables in addition to the multibody data enables optimization loops between multibody simulation and the related CAD model (Figure 1). An example demonstrates the three main steps of the presented method, divided into model processing, package generation and parameterized package update. Furthermore, the update process is integrated in a manual parameter variation as well as an automated optimization routine to enable parametric design studies coupled with multibody simulation.

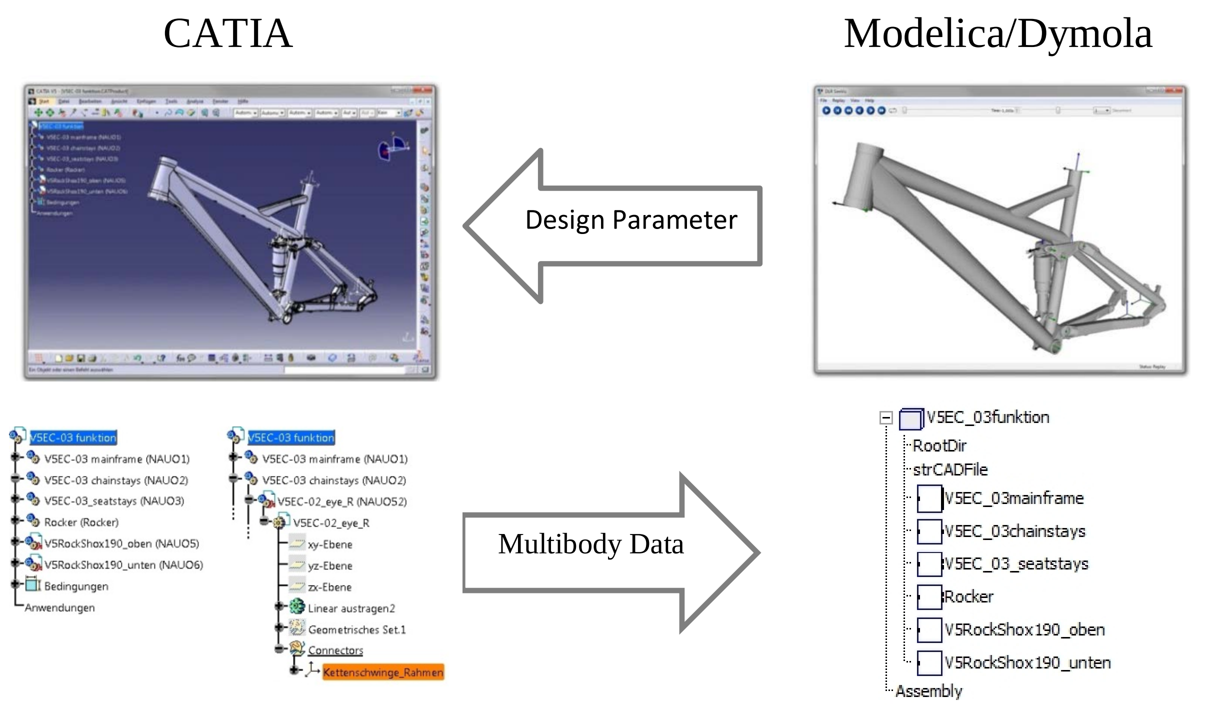


Figure 1: Automated package generation and parametric design

Modelling elastomer buffers with DyMoRail

Elisabeth Dumont Werner Maurer
Institut für Angewandte Mathematik und Physik
Zürcher Hochschule für Angewandte Wissenschaften
Technikumstrasse 9, Winterthur, 8401 Switzerland

In this paper a model for elastomer buffers for longitudinal railway vehicle dynamics is presented. This model is part of the more extended DyMoRail library which allows to simulate longitudinal dynamics of entire railway trains. With this library an efficient simulation of complete train compositions in various combinations is possible. The elastomer buffer can be used in combination with other buffer models and couplers in different test scenarios. We present details of our rubber spring model based on the one-dimensional, non-linear rubber spring model proposed by M. Berg [1][2][3]. To illustrate the behavior of the friction force modelled in the latter, it is compared to a diode model for Coulomb friction similar to the one in the Modelica Standard Library. Simulations for 40 J-buffer known as ”Miner40” used for freight waggons during shunting at speeds up to 12 km/h are shown. Also shown are simulations of an entire S-Bahn combination with sixteen cars and fifteen elastomer buffers.

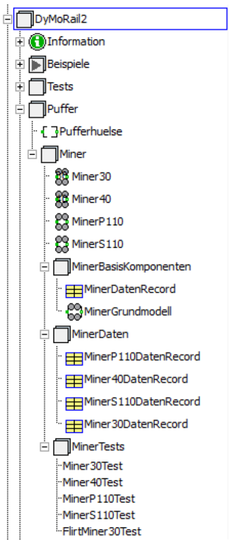


Figure 1: Structure of the buffer sublibrary ”Puffer”.

References

[1] Berg, M. A model for rubber springs in the dynamic analysis of rail vehicles, Proc Instn Mech Engrs, Vol 211 Part F (1997), pp. 95-108.

[2] Berg, M. A Non-Linear Rubber Spring Model for Vehicle Dynamics Analysis, Vehicle System Dynamics Supplement, Swets and Zeitlinger Publishers, 28 (1998), pp. 723-728.

[3] Berg, M. A Non-Linear Rubber Spring Model for Rail Vehicle Dynamics Analysis, Vehicle System Dynamics, Swets and Zeitlinger Publishers, 30 (1998), pp 197 - 212.

A Modelica Contact Library for Idealized Simulation of Independently Defined Contact Surfaces

Felix Oestersötebier Peng Wang Ansgar Trächtler
Control Engineering and Mechatronics,
Heinz Nixdorf Institute, University of Paderborn
Fürstenallee 11, 33102 Paderborn, Germany
{felix.oestersoetebier, peng.wang, ansgar.traechtler}@hni.uni-paderborn.de

Considering technical use-cases one often finds multibody mechanics that typically comprise mainly fixed (e.g. kinematic chains) and few loose couplings, where dynamic contact phenomena take place. Nevertheless, modeling contacts is “a key factor and a challenging problem in simulation of multibody systems, where a balance between performance and accuracy has to be found” [1]. However, to the best of our knowledge, there is currently no Modelica library available to handle contact problems in any level of detail. The library presented in this paper implements contact modeling by means of non-central contact blocks. It also provides ready-to-use blocks for simple contact surfaces. The surface blocks can be connected to any kind of rigid body by a *frame* connector and the dimensions of the surface can be parameterized. We introduce a new interface to connect the surface definition with the contact blocks, which makes the surface dimensions available for the contact calculation. The overall aim is to enable the designer of mechatronic systems to perform simulations of systems including idealized representation of contacts. Therefore, we use an idealization of the so called “foundation model” [2] that is based on single force elements. Assuming that the contact area is not only relatively small but also of an idealized shape, we describe it by means of single contact points. A nonlinear spring-damper element is inserted to calculate the normal force between these points of the colliding surfaces. The normal force is then used to determine the friction force between the two bodies by a continuous approximation of the Stribeck curve. This requires the previous identification of potential contact points on the rigid body surfaces and the continuous determination of the normal direction. For these purposes, we provide analytic solutions for simple geometries in our library. As a starting point, we focused on *spherical*, *cylindrical* and *plane* surfaces, either in *rectangular* or *circular* shape. Depending on the shape of the contact area, we use 1 (punctiform), 2 (linear) or 4-5 (planar) points to describe it. Exemplarily, the results of three experiments are shown and compared to benchmark simulations in RecurDyn. They demonstrate that the idealized contact library provides a powerful and easy to use opportunity to model contact phenomena of simple contact geometries. After making our library publicly available, we hope to identify further opportunities for improvement with the help of the Modelica community in the future.

This work was developed in the project “ENTIME: Entwurfstechnik Intelligente Mechatronik” (Design Methods for Intelligent Mechatronic Systems). The ENTIME project was funded by the state of North Rhine-Westphalia (NRW), Germany, and the EUROPEAN UNION, European Regional Development Fund, “Investing in your future”.

References

[1] T. Juhász. Advanced Solutions in Object-Oriented Mechatronic Simulation. Ph.D. Thesis, Dept. of Control Engineering and Information Technology, Budapest University of Technology and Economics, 2008

[2] K. L. Johnson. Contact Mechanics. Cambridge University Press, Cambridge, 1985

The OneWind® Modelica Library for Wind Turbine Simulation with Flexible Structure — Modal Reduction Method in Modelica

Philipp Thomas Xin Gu Roland Samlaus Claudio Hillmann Urs Wihlfahrt
Fraunhofer Institute for Wind Energy and Energy System Technology IWES
Am Seedeich 45, 27572 Bremerhaven, Germany

The OneWind® Modelica Library[1] [1] for coupled wind turbine loads calculation developed at Fraunhofer IWES uses a structural element based on a modal reduction method to model the motion and deformation of flexible wind turbine rotor blades and tower. The degrees of freedom (DOF) are rigid body motions and modal DOF. The `ModalElement` model allows the simulation of coupling effects like bend-twist coupling in wind turbine rotor blades and the structural behavior is dependent on the selected eigenmodes. This paper gives an overview about the Modelica implementation of the theory of modal elements, the advantages over other methods (finite-elements), how the `ModalElement` model is included into the OneWind® Modelica Library, and how it is used for load calculation.

The objective is to achieve the same accuracy as the finite element method (see Figure 1) with less computational time. This is a requirement for a wind turbine load calculation with a large number of load cases which are necessary for fatigue analysis.

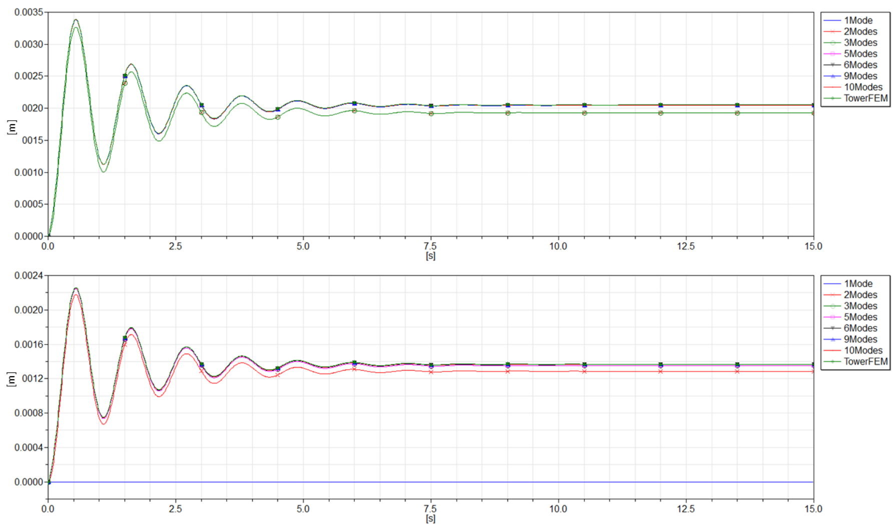


Figure 1: Deflection of `ModalElement` model based wind turbine tower top: upper plot shows downwind displacement, lower plot shows side-to-side displacement

References

[1] Strobel M., Vorpahl R., Hillmann C., Gu X., Zuga A., Wihlfahrt U. The OnWind Modelica Library for offshore wind turbines – Implementation and first results. In: Proceedings of the 8th International Modelica Conference 2011, Dresden, Germany, Modelica Association, pp. 603 - 609, 20-22 March 2011.

[1] Version 1.0 for onshore wind turbine load calculation was released and is now available under dual license model. For further information contact: info@onewind.de

Simulating Collisions within the Modelica MultiBody Library

Andreas Hofmann[1] Lars Mikelsons[1] Ines Gubsch[2] Christian Schubert[2]
[1]Bosch Rexroth AG, Lohr am Main, Germany
{Andreas.Hofmann7, Lars.Mikelsons}@boschrexroth.de
[2]TU Dresden, Chair of Construction Machines and Conveying Technology, Germany
{ines.gubsch, christian.schubert}@tu-dresden.de

Abstract

In this paper an approach for handling collision within the Modelica MultiBody library is presented. Therefore, a short overview about collision consideration for multibody simulation is given. Different methods for calculating the contact reactions are discussed and their potentials for implementation in a free Modelica library are deliberated. Furthermore the implementation of this collision library, using a penalty-based collision approach and the *Bullet Physics Library* for collision detection is described. The application is demonstrated in examples and limitations are brought up. Although some drawbacks restrict usability, the library can be used to increase the level of detail for multibody simulation models.

Introduction

Many physical systems cannot be simulated in a feasible manner without the description of collision interaction. Not only the typical applications, like wheel-road- contact, newton's cradle or a bouncing ball need collision consideration, but especially real-life models require contact handling. For example, simulation of typical working processes for construction machines, with lifting rocks can benefit from this. But also machine elements like mechanical springs require collision handling for simulations including dynamic loads. For the Modelica Library Modelica. Mechanics.MultiBody (M.MB) several collision handling considerations have been made, with two to be shortly mentioned. In [1] Otter et al. introduced an extension to the M.MB library with capabilities of handling collisions. Collision detection using different approaches of surface representation were shown. Engelson [2] described a way of contact implementation using impulse-based and penalty-based methods. However, those approaches have never been available in public. To offer collision handling to general public, CollisionLib – the library presented here – will be freely available. Although the functionality of this very first version has only been tested in Dymola, support for OpenModelica and other Modelica environments are planned for the future.

References

[1] Otter, M. et al. Collision Handling for the Modelica MultiBody Library. In: Proceedings of the 4th Modelica Conference 2002, Hamburg, Germany, Modelica Association, 7-8 March 2005.

[2] Engelson, V. Integration of Collision Detection with Multibody System Library in Modelica. Linköping, Sweden: Thesis, Department of Computer and Information Science, Linköping University, 2000.

Short-term production planning for district heating networks with JModelica.org

Stéphane Velut[1], Per-Ola Larsson[1], Johan Windahl[1], Linn Saarinen[2], Katarina Boman[2]

[1] Modelon AB, Ideon Science Park, SE-223 70 Lund, Sweden

[2] Vattenfall R&D, 814 26 Älvkarleby, Sweden

stephane.velut@modelon.com, per-ola.larsson@modelon.com

The short term thermal production planning problem is solved in two steps by integrating physical plant models into the standard approach. The first step aims at solving the discrete variables from the unit commitment sub-problem (UCP) using standard mixed integer linear models and optimization techniques. The second step focuses on the economic dispatch sub-problem (EDP) described by high-fidelity, continuous time, physics-based Modelica models together with nonlinear optimization techniques from the JModelica.org platform [1]. The output of the second step includes optimized power flows but also highly relevant variables such as supply temperature, supply flow rate, turbine by-pass valve in the cogeneration plant. The optimization is formulated as a maximization of the benefit from heat and electricity sell over a finite time-horizon.

The proposed method is validated in several test cases using experimental data from a plant in Nyköping. The optimizations demonstrate the feasibility and the high economic potential of the proposed approach when comparing with measurement data and the standard optimization techniques. The optimized planning schedules result in a balance between produced and consumed heat, priority to low-cost boilers and maximization plant revenue. Compared to measurement data, the optimizations result in a significantly lower supply temperature, a more extensive usage of the external cooler for higher efficiency and higher electricity production, fewer starts of units as well as an appropriate use of the accumulator tank.

The high-level description of optimization problems using JModelica.org provides useful means to specify flexible optimization problems including constraints on arbitrary process variables such as heat load of the production units, supply temperature and flow rate, pressures.

References

[1] Åkesson J., Årzén K-E., Gäfvert M., Bergdahl T., Tummescheit H.: Modeling and Optimization with Optimica and JModelica.org—Languages and Tools for Solving Large-Scale Dynamic Optimization Problems, Computers and Chemical Engineering, 34:11, pp. 1737–1749, November 2010.

Modelling the system dynamics of islanding asynchronous generators

Håkon Molland Edvardsen
Oslo, Norway
haakky1@gmail.com

Dietmar Winkler
Telemark University College, Norway
dietmar.winkler@hit.no

Asynchronous generators are often used for small hydro power stations with an installed power capacity of under $1\,MW$. The reason for this is their robustness and low cost. In order do be able to produce active electrical power with an asynchronous generator once needs to provide enough excitation by means of reactive power provided by either the electrical grid or additional capacitors.

But in asynchronous generators we can also find the phenomenon of self-excitation which allows the asynchronous generator to operate as a standalone unit. Investigation of the self-excitation process shows that significant over-voltages can occur if a generator with sufficient capacitors is suddenly disconnected from the utility grid. The precondition for a successive voltage build-up is that the generator is left with enough capacitive power and a low load after the disconnection.

The *Lønnestad* radial in *Seljord, Norway*, is a distribution radial with both asynchronous and synchronous generators connected. In order to investigate the system dynamics in the radial after it is disconnected from the rest of the $22kV$ distribution grid, the radial was modelled and simulated using Modelica as modelling language.

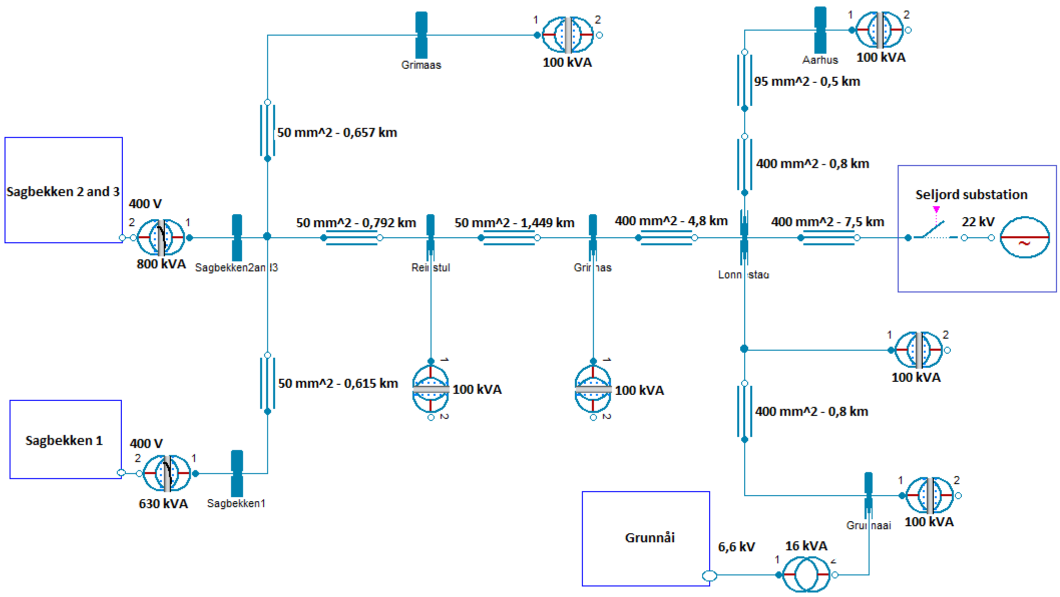


Figure 1: Overview of the Lønnestad radial

Hybrid Energy System Modeling in Modelica

William R. Binder[1] Christiaan J. J. Paredis[1] Humberto E. Garcia[2]
[1]Georgia Institute of Technology
813 Ferst Drive, Atlanta, GA 30332, USA
binderw@gatech.edu, chris.paredis@me.gatech.edu
[2]Idaho National Laboratory
2525 N. Fremont Drive, Idaho Falls, ID 83415-3675, USA
Humberto.Garcia@inl.gov

In this paper, a Hybrid Energy System (HES) configuration is modeled in Modelica. Hybrid Energy Systems (HES) have as their defining characteristic the use of one or more energy inputs, combined with the potential for multiple energy outputs. Compared to traditional energy systems, HES provide additional operational flexibility so that high variability in both energy production and consumption levels can be absorbed more effectively. This is particularly important when including renewable energy sources, whose output levels are inherently variable, determined by nature.

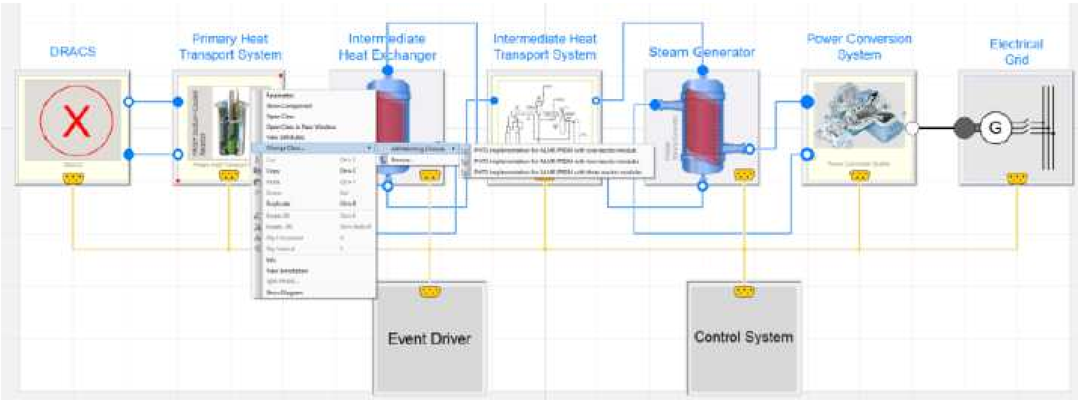
The specific HES configuration modeled in this paper include two energy inputs: a nuclear plant, and a series of wind turbines. In addition, the system produces two energy outputs: electricity and synthetic fuel. The models are verified through simulations of the individual components, and the system as a whole. The simulations are performed for a range of component sizes, operating conditions, and control schemes.

Keywords: hybrid energy system; Modelica; multiple-input; multiple-output; renewable power; optimization

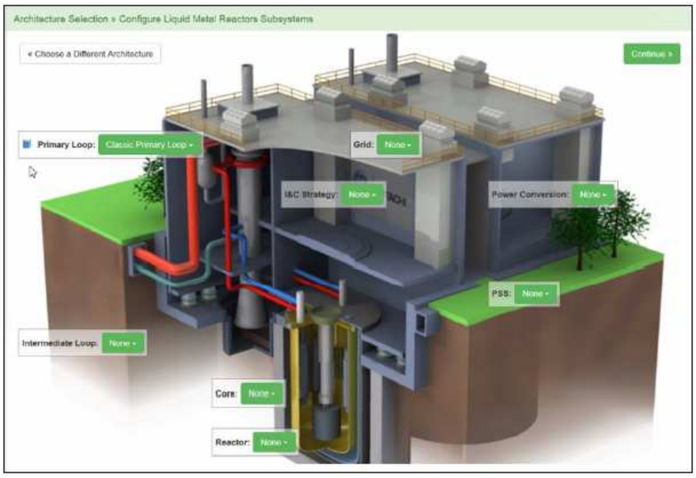
Dynamic Modeling of Small Modular Nuclear Reactors using MoDSim

Richard Hale[1] Sacit Cetiner[1] David Fugate[1] Lou Qualls[1]
John Batteh[2] Michael Tiller[3]
[1]Oak Ridge National Labs, Oak Ridge, TN USA
[2]Modelon, Inc., Ann Arbor, MI USA
[3]Xogeny, Inc., Canton, MI USA
{halere1, cetinerms, fugatedl, quallsal}@ornl.gov
john.batteh@modelon.com michael.tiller@xogeny.com

As part of the advanced small modular nuclear reactor (AdvSMR) R&D program, Oak Ridge National Laboratory (ORNL) is developing a Dynamic System Modeling Tool (MoDSim) to facilitate research and development related primarily to instrumentation and controls (I&C) studies of small modular reactors (SMRs). This paper describes the use of Modelica models and Functional Mockup Interface (FMI) to model, simulate, and deploy compiled models via a web-based platform based on FMQ from Xogeny. The web platform allows both model configuration and parameter specification. The application employs the FMI Add-in for Excel from Modelon to provide automated simulation and post-processing capability locally in Microsoft Excel. The key features of the models, application to the PRISM reactor concept, and simulation platform are discussed and initial results presented.



SMR modeling architecture



SMR model configuration via web-based platform

Modified Multiple Shooting Combined with Collocation Method in JModelica.org with Symbolic Calculations

Evgeny Lazutkin, Abebe Geletu, Siegbert Hopfgarten, Pu Li
Simulation and Optimal Processes Group
Institute for Automation and Systems Engineering,
Technische Universität Ilmenau, P.O.Box 10 05 65, 98684 Ilmenau, Germany.
(evgeny.lazutkin, abebe.geletu, siegbert.hopfgarten, pu.li)@tu-ilmenau.de

This paper presents an efficient and novel implementation of a combined multiple shooting and collocation (CMSC) algorithm for the solution of nonlinear optimal control problems (NOCP). The implemented algorithm is a modification of the approach proposed in [1, 2]. The new implementation is done under the JModelica.org framework along with CasADi and Ipopt. The framework uses a symbolic pre-calculation of functions and derivatives. The paper gives a description of the algorithm and elaborates the components of the framework. Comparative numerical experimentations show that the new implementation is efficient in comparison with the published results of other authors.

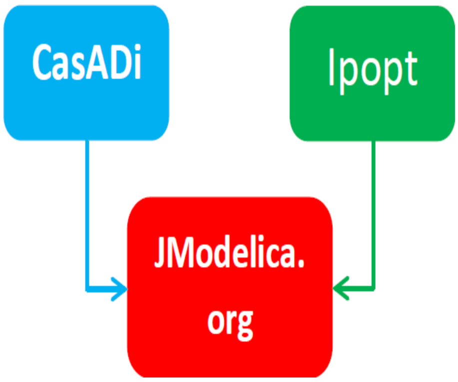


Figure 1: Software framework

The major difference from the original version of the CMSC approach in [1, 2] consists in the use of pre-calculated derivatives and their symbolic representation. That is, in every iteration of the optimization algorithm, sensitivities are automatically available without further calculations. This leads to accurate results with outstanding speedup of the overall computation time. In the framework shown in Fig. 1, the NOCP is modeled under JModelica.org using a Python script. Then the problem is discretized using our CMSC with the help of CasADi to obtain a nonlinear programming problems (NLP). Subsequently, CasADi is again invoked to generate symbolic expressions for the derivatives. Finally, JModelica.org invokes Ipopt to solve the NLP by using the pre-calculations.

References

[1] Tamimi, J.: Development of the Efficient Algorithms for Model Predictive Control of Fast Systems. PhD Thesis, Technische Universität Ilmenau, VDI Verlag, 2011.

[2] Tamimi, J., Li, P.: A Combined approach to nonlinear model predictive control of fast systems. J. Process Control, 20(2010)9, pp. 1092-1102.

DOML - a Compiler Environment for Dynamic Optimization Supporting Multiple Solvers

Tomasz Tarnawski Radosław Pytlak
Warsaw Technical University, Institute of Automatic Control and Robotics
ul. sw Andrzeja Boboli 8, 02-525 Warsaw
{tarnawski,pytlak}@mchtr.pw.edu.pl

The Modelica language may serve well as a base for defining optimal control problems, given a few relatively minor syntax extensions. One example proving that point is Optimica (initially proposed in [1]) and another one is DOML (Dynamic Optimization Modeling Language) – installed on IDOS (Interactive Dynamic Optimization Language, on–line at [4]) and described in the paper. The DOML implementation is, actually, heavily based on the (open source) compiler of Optimica (available at [3]) but it provides a number of important features absent in its precursor.

One main extension of the compiler lies in its built-in mechanism supporting the use of many different optimization solvers (selected on the fly, depending on the content of the problem definition) and to seamlessly add new, external, solvers. We find the need for implementing support for multiple solvers to be justified and intuitive. For once, dynamic optimization problems come in many different kinds (e.g. parametric, minimum time, DAE with higher index, etc) and so one solver would not be general enough to solve all of them efficiently and accurately; solvers dedicated to particular problem kinds simply do the job better. The second reason is to open the possibility for applying "solver chaining" i.e. a strategy of using two (or more) solvers consecutively on one problem. The first one yields a crude approximation while the next guarantees a more accurate solution but requires a reasonably good initial guess (and possibly other warm-starting information) – that may be taken form the first solver. For more details see e.g. [2].

In result, the range of problems that may be specified with DOML and solved in the IDOS environment is quite wide and keeps growing. The scope of problem ranges from some static optimization problems through regular ODE, parametric optimization, minimum time problems, up to DAE with higher index. DOML language extensions also provide preliminary support for multi-objective optimization and PDE problems.

References

[1] Åkesson J. Tools and Languages for Optimization of Large-Scale Systems. Lund, Sweden: *PhD thesis*, Department of Automatic Control, Lund University, 2007.

[2] Pytlak R, Błaszczyk J, Krawczyk K, Tarnawski T. Solvers chaining in the IDOS server for dynamic optimization. In: Proceedings of 52nd Conference on Decision and Control, Florence, Italy, 10-13 December 2013.

[3] Open Source Project: JModelica.org. `http://www.jmodelica.org/`. November 2013.

[4] Interactive Dynamic Optimization Server. `http://idos.mchtr.pw.edu.pl/`. November 2013.

Efficient Implementation of Collocation Methods for Optimization using OpenModelica and ADOL-C

Vitalij Ruge Willi Braun Bernhard Bachmann
{vitalij.ruge, willi.braun, bernhard.bachmann}@fh-bielefeld.de
Bielefeld University of Applied Sciences, Department of Mathematics and Engineering,
Bielefeld, Germany

Andrea Walther Kshitij Kulshreshtha
andrea.walther@uni-paderborn.de kshitij@math.upb.de
Universität Paderborn, Institut für Mathematik,
Paderborn, Germany

Abstract

Efficient calculation of the solutions of nonlinear optimal control problems (NOCPs) is becoming more and more important for today's control engineers. The systems to be controlled are typically described using differential-algebraic equations (DAEs), which can be conveniently formulated in Modelica. In addition, the corresponding optimization problem can be expressed using Optimica.
Solution algorithms based on collocation methods are highly suitable for discretizing the underlying dynamic model formulation. Thereafter, the corresponding discretized optimization problem can be solved [1], e.g. by the interior-point optimizer Ipopt. The performance of the optimizer heavily depends on the availability of derivative information for the underlying optimization problem.
Typically, the gradient of the objective function, the Jacobian of the DAEs as well as the Hessian matrix of the corresponding Lagrangian formulation need to be determined. If only some or none of these derivatives are provided, usually numerical approximations are used by the optimizer internally.

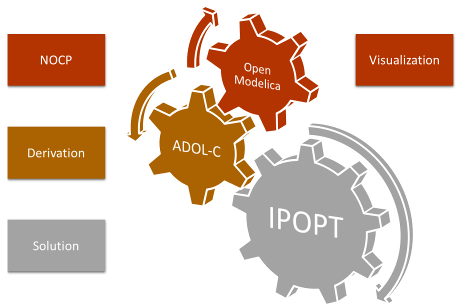


Figure 1: Structure of the coupling

OpenModelica supports the Optimica language and is capable of automatically generating the discretized optimization problem using collocation methods as well as the whole symbolic machinery available. In addition, all necessary derivative information is determined using the automatic differentiation capabilities of ADOL-C, which has now been integrated into the OpenModelica environment. The performance of the new developed tool chain 1 is demonstrated on a more industry relevant benchmark, which is a model of a combined cycle power plant model [2], see figure 2.

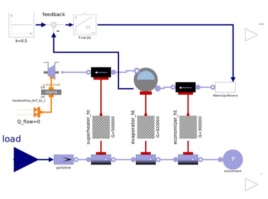


Figure 2: CombinedCycle

References

[1] B. Bachmann et al.. Parallel Multiple-Shooting and Collocation Optimization with OpenModelica. 9th International Modelica Conference, 2012.

[2] F. Casella, F. Donida and J. Åkesson. Object-Oriented Modeling and Optimal Control: A Case Study in Power Plant Start-Up. Lund, 8th IFAC World Congress, Milano, Italy, 2011.

Symbolic Transformations of Dynamic Optimization Problems

Fredrik Magnusson[a] Karl Berntorp[a] Björn Olofsson[a] Johan Åkesson[a,b]
[a]Department of Automatic Control, Lund University, SE-221 00 Lund, Sweden
[b]Modelon AB, Ideon Science Park, SE-223 70 Lund, Sweden

Dynamic optimization problems occur in various fields and applications. These include parameter estimation and optimal control. Optimization problems involving differential-algebraic equation (DAE) systems are traditionally solved while retaining the semi-explicit or implicit form of the DAE. We instead consider symbolically transforming the DAE into an ordinary differential equation (ODE) prior to solving the optimization problem using a collocation method. We present a method for achieving this that treats DAE-constrained optimization problems. The method is based on techniques commonly used in Modelica tools for simulation of DAE systems. These techniques involve causalization of DAE systems and transformation to block-lower triangular form of the equations.

The method has been implemented in the JModelica.org open-source Modelica platform. The implementation is evaluated on two industrially relevant benchmark problems. The first problem is a minimum-time problem, where we seek the time-optimal maneuver for an automobile in a hairpin turn, see Figure 1. The second problem is to optimize the warm startup of a combined-cycle power plant, whose model diagram is displayed in Figure 2.

Figure 1: An example of a hairpin turn. Photo courtesy of RallySportLive.

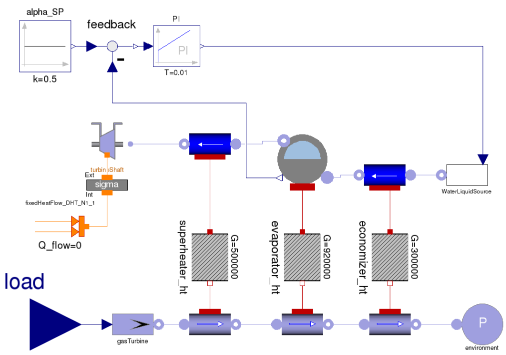


Figure 2: Power plant model diagram

The problems are solved using both the traditional DAE formulation and the proposed ODE formulation. The performance of the two approaches is compared. The ODE formulation is shown to have between two and four times shorter online execution time for the considered cases. We also discuss benefits and drawbacks of the two approaches.

Modelling a Lignite Power Plant in Modelica to Evaluate the Effects of Dynamic Operation and Offering Grid Services

M. Hübel[2], A. Berndt[2], S. Meinke[1], M. Richter[2], P. Mutschler[2],
E. Hassel[2], H. Weber[2], M. Sander[2], J. Funkquist[1]
[1] Vattenfall Research and Development
Otternbuchtstrasse 14-16, 13599 Berlin, Germany
[2] University of Rostock
A.-Einstein-Str. 2, 18059 Rostock, Germany
<sebastian.meinke@vattenfall.de> <moritz.huebel@uni-rostock.de>

Offering services to stabilize the electrical grid is one of the major tasks of fossil power plants and also of significant economical relevance. However the effects on the power plants regarding the additional wear of components is uncertain. Usually the effects regarding control reserves, especially primary control occur with high frequencies and small amplitudes, which makes investigations based on measurement data impossible since the effects are masked by the noise of normal operation. In order to investigate this issue, a detailed model of a lignite power plant has been used, which was developed in Modelica for simulating and comparing scenarios with and without offering primary control reserves. The model comprises the entire water-steam cycle including turbines, preheaters and pumps, as well as a very detailed boiler model including the air supply, coal mills, a combustion chamber, heating surfaces and piping. Furthermore the power plants control system has been implemented in a very precise way. In addition the study involves an investigation on the input signals (grid frequency) and a calculation of lifetime consumption for specific components to evaluate the effects.

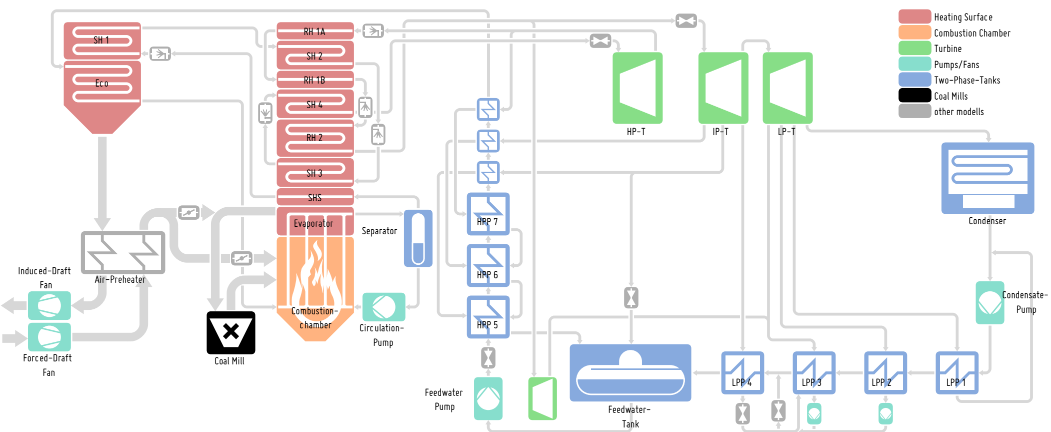


Figure 1: Implemented components of reference power plant

Use of External Fluid Property Code in Modelica for Modelling of a Pre-combustion CO_2 Capture Process Involving Multi-Component, Two-Phase Fluids

Carsten Trapp *[1], Francesco Casella[2], Teus van der Stelt[1] and Piero Colonna[1]

[1]*Delft University of Technology, Propulsion and Power, Kluyverweg 1, 2629 HS Delft, The Netherlands*
[2]*Politecnico di Milano, Dipartimento di Elettronica, Informazione e Bioingegneria, Via Ponzio 34/5, 20133 Milano, Italy*

Pre-combustion CO_2 capture applied to integrated gasification combined cycle (IGCC) power plants is a promising technical solution to mitigate CO_2 emissions and therefore the effect of climate change. The integration of the CO_2 removal unit with the very complex gasification process and combined cycle power plant leads to challenges especially regarding dynamic operation, which nowadays becomes increasingly important as the share of electricity produced by renewable energy sources, which is inherently unsteady, is continuously growing.

This paper presents the development of a system model for a pre-combustion CO_2 capture process in order study process transient performance during load variations. The models have been validated by comparison with experimental data obtained from a unique, fully instrumented CO_2 capture pilot plant, which has been realized at the Buggenum IGCC power station in the Netherlands by the utility company Vattenfall.

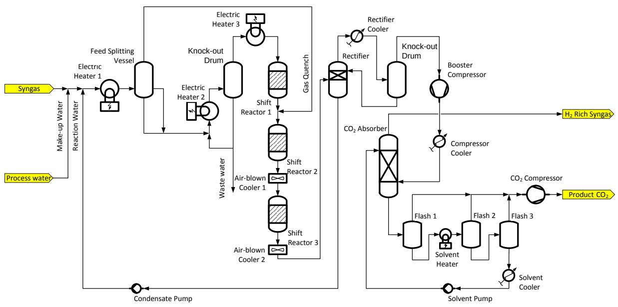


Figure 1: Process flow diagram CO_2 capture pilot plant.

The main challenge of the model development is related to the computation of fluid properties, in particular phase equilibria, due to the fact that highly non-ideal, two-phase, multi-component fluids are involved in the capture process, which are currently not supported by available Modelica media libraries or interfaces.

Therefore, an interface prototype was developed and tested with the fluid property package FluidProp for the modelling and simulation of the CO_2 capture process. Due to limitations regarding index reduction if using external property functions, an approach on how to develop index-1 models and choose system state variables as well as thermodynamic states appropriately is discussed. Furthermore, various ideas in order to improve computational efficiency, when expensive phase equilibria calculations are involved, are presented. Recommendations about the design of a library for the use of external property estimation code in Modelica conclude the treatment.

*Email: C.Trapp@tudelft.nl, P.Colonna@tudelft.nl

Dynamic modelling of a parabolic trough solar power plant

Robert Österholm[a] , Jens Pålsson[b]
[a]Lund University, LTH, Department of Energy Sciences, Lund, Sweden
[b]Modelon AB, Ideon Science Park, Lund, Sweden

osterholm.r@gmail.com, jens.palsson@modelon.com

Abstract

Models for dynamic simulation of a parabolic trough concentrating solar power (CSP) plant were developed in Modelica for the simulation software tool Dymola. The parabolic trough power plant has a two-tank indirect thermal storage with solar salt for the ability to dispatch electric power during hours when little or no solar irradiation is present. The complete system consists of models for incoming solar irradiation, a parabolic trough collector field, thermal storage and a simplified Rankine cycle. In this work, a parabolic trough power plant named Andasol located in Aldeire y La Calahorra, Spain is chosen as a reference system. The system model is later compared against performance data from this reference system in order to verify model implementation. Test cases with variation in solar insolation reflecting different seasons is set up and simulated. The tests show that the system model works as expected but lack some of the dynamics present in a real thermal power plant. This is due to the use of a simplified Rankine cycle. The collector and solar models are also verified against literature regarding performance and show good agreement.

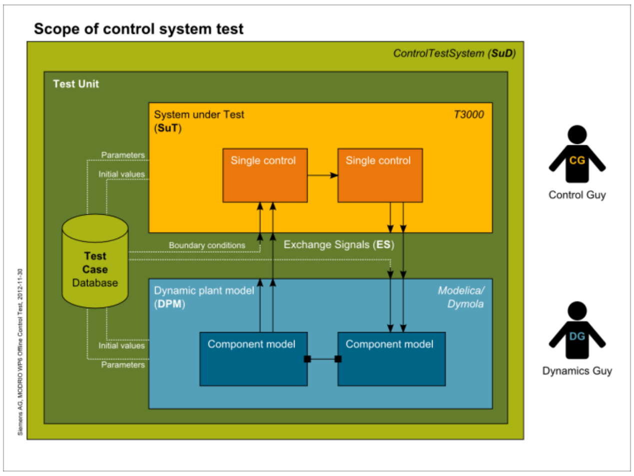
Keywords: concentrating solar power; parabolic trough; solar salt thermal storage; dynamic modeling; Dymola; Modelica.

Testing Power Plant Control Systems in Modelica

Kilian Link[a], Leo Gall[b]
Julien Bonifay[a], Matthias Buggert[a]
[a]Siemens AG, Energy Sector, Erlangen, Germany
[b]Bausch-Gall GmbH, Munich, Germany

The control test application shown in this paper demonstrates the usability of Modelica to run comprehensive tests for plant controllers involving large parts of the physical plant and the control system. The goal is to allow the easy testing of controllers for power plants developed in a distributed control systems (DCS) such as SPPA-T3000 at every control engineer's desk. Due to the urgent need for control testing in a running project, this fully Modelica based prototype developed in the MODRIO project [2] has then be applied to a real life example. Similar to the approach of [1], the behavior of control blocks has been re-implemented in Modelica. A customized T3000-Modelica parser has been developed which enables the control engineer to translate a set of functional block diagrams from SPPA-T3000 into a Modelica package. This package contains all functional block diagrams which are to be tested and a top-level class containing all open controller inputs.

As a first application, the control of a district heating system of a combined cycle plant (CCPP) was investigated. The district heating system uses steam from steam turbine extractions in order to heat the water that will be supplied for district heating. The main function of the controller is to control the supply temperature of the water by means of several valves in the plant. A district heating system has long piping connections which are modeled as transmission-line pipe models with spatial distribution for the enthalpy.



The controller model (System under Test) and the plant model are connected in one Modelica model, called Test Unit. The Figure on the left shows the current approach for signal handling. Besides connecting controller and plant model, the Test Unit is used for adding boundary conditions. Test Cases are then created by extending the Test Unit and adding disturbances and changing parameters.

This paper shows that the simulation based test of control systems in Modelica is possible for industrial use cases. One of the identified benefits of the self-contained simulation setup was that absolutely no changes to the authoring tool (SPPA-T3000) were necessary. Nevertheless, we are reaching the limits of what current Modelica tools can handle in terms of number of signals and simulation performance of hybrid systems.

References

[1] Marco Bonvini, Alberto Leva (2012). A Modelica Library for Industrial Control Systems, Proceedings of the 9th International MODELICA Conference, September 3-5, 2012, Munich, Germany

[2] MODRIO - Model Driven Physical Systems Operation, http://www.modrio.org

Vehicle Thermal Management – A Case Study in Web-Based Engineering Analysis

Michael Tiller
Xogeny Inc., USA
michael.tiller@xogeny.com

Modelica has proven to be a compelling technology for creating sophisticated multi-domain models. It provides modern language features to promote model reuse and maximize developer productivity. These capabilities are backed up by proven simulation performance. More recently, the Functional Mockup Interface standard (FMI) has created an avenue for these models to be exported outside the model development environment as Functional Mockup Units (FMUs).

In this paper, we explore one possible way to utilize models that have been exported as FMUs. Specifically, we discuss how to incorporate these models into a web-based engineering analysis application that is designed to be accessible to non-expert users. Our goal is to show the important role that web and cloud based approaches can have in magnifying the impact of modeling activities across the enterprise.

We consider a specific engineering model and discuss exactly how we have transformed the model to make it accessible as a web-based application. This includes a discussion of the input and output data associated with the model as well as how a web based deployment (backed by cloud based services) can provide unique features compared to more conventional methods of model deployment.

recon – Web and network friendly simulation data formats

Michael Tiller
Xogeny Inc., USA
michael.tiller@xogeny.com

Peter Harman
CyDesign Ltd., UK
peter@cydesign.com

There are many different commonly used file formats for storing time series data. Most of these file formats are designed with the assumption that the file itself will be locally available to the software that will be reading or writing the data stored in them. While this assumption is an excellent one for desktop based tools with direct access to disk drives capable of moving virtually instantaneously around from sector to sector, there are a growing number of applications for which local access is not necessarily available. For these applications, we've initiated the `recon` project to develop more suitable formats.

With the emergence of web and cloud based modeling and simulation technologies, the time has come to explore file formats specifically optimized for non-desktop applications. In this paper, we present a new set of file formats that are specifically designed for web and cloud based approaches. This paper reviews the key requirements for web and cloud enabled applications and then presents a specification for two file formats that address those requirements.

When considering the various use cases that drove our requirements, we recognized that two different file formats were actually required. The first format, the `wall` format, is optimized for writing. The other format, the `meld` format, is optimized for reading over a network (*i.e.,* minimizing the number of reads and bytes read). We will discuss the layout of each of these formats and describe the use cases for which they are most appropriate.

In the open tradition of the Modelica Association, the authors have made specifications and implementations for these formats available as open source libraries with the hope that they will benefit the community as a whole.

IDOS - (also) a Web Based Tool for Calibrating Modelica Models

Radosław Pytlak Tomasz Tarnawski
Warsaw Technical University, Institute of Automatic Control and Robotics
ul. sw Andrzeja Boboli 8, 02-525 Warsaw
{pytlak,tarnawski}@mchtr.pw.edu.pl

This paper presents a newly deployed server, `IDOS`, an online-accessible environment providing the service of solving optimal control problems ([2],[3],[5]). Development and deployment of the Interactive Dynamic Optimization Server is a result of projects funded by NCBiR (National Center for Research and Development). One of the outcomes of the project was a modeling language (Dynamic Optimization Modeling Language, `DOML`) providing a uniform format for defining dynamic optimization problems. `DOML` is an extension of `Modelica` language ([4]) and hence, not only a user can specify his problem in the way he does in `Modelica` but also (more importantly, for the purpose of this paper) models created in `Modelica` for simulation purposes can be easily transferred to `DOML` for solving their related optimization problems. In particular, `Modelica` models can be calibrated with the help of our server. The paper tries to illustrate the point in depth. It presents the workings of the server and reviews the scope of solvers implemented, focusing especially on those that can be used for calibrating `Modelica` models. Special attention is devoted to an algorithm using adjoint equations for evaluating sensitivities of model equations with respect to parameters and to calibrating models described by higher index DAEs ([1]).

References

[1] Pytlak R. Numerical procedure for optimal control of higher index DAEs. In: J. Discrete and Continuous Dynamical Systems, Vol. 29, 1–24, 2011.

[2] Pytlak R, Tarnawski T, Fajdek B, Stachura M. Interactive dynamic optimization server–connecting one modeling language with many solvers. In: Optimization Methods and Software, 2013. http://dx.doi.org/10.1080/10556788.2013.799159.

[3] Pytlak R, Błaszczyk J, Krawczyk K, Tarnawski T. Solvers chaining in the IDOS server for dynamic optimization. In: Proceedings of 52nd Conference on Decision and Control, Florence, Italy, 10-13 December 2013.

[4] Open Source Project: JModelica.org. http://www.jmodelica.org/. November 2013.

[5] Interactive Dynamic Optimization Server. http://idos.mchtr.pw.edu.pl/. November 2013.

Client-side Modelica powered by Python or JavaScript

Rüdiger Franke, ABB, Germany – Ruediger.Franke@de.abb.com

Modelica is primarily supported by simulation environments for the treatment of equation based models and model libraries. As of today Modelica is rarely used for the exchange of engineering data, visualization or interactive computing, even though the Modelica language offers a lot of interesting features for such applications.

This paper investigates the potential of lightweight Modelica tools that run directly in scripting or web clients. Two Modelica parsers have been implemented in the popular client-side languages Python and JavaScript.

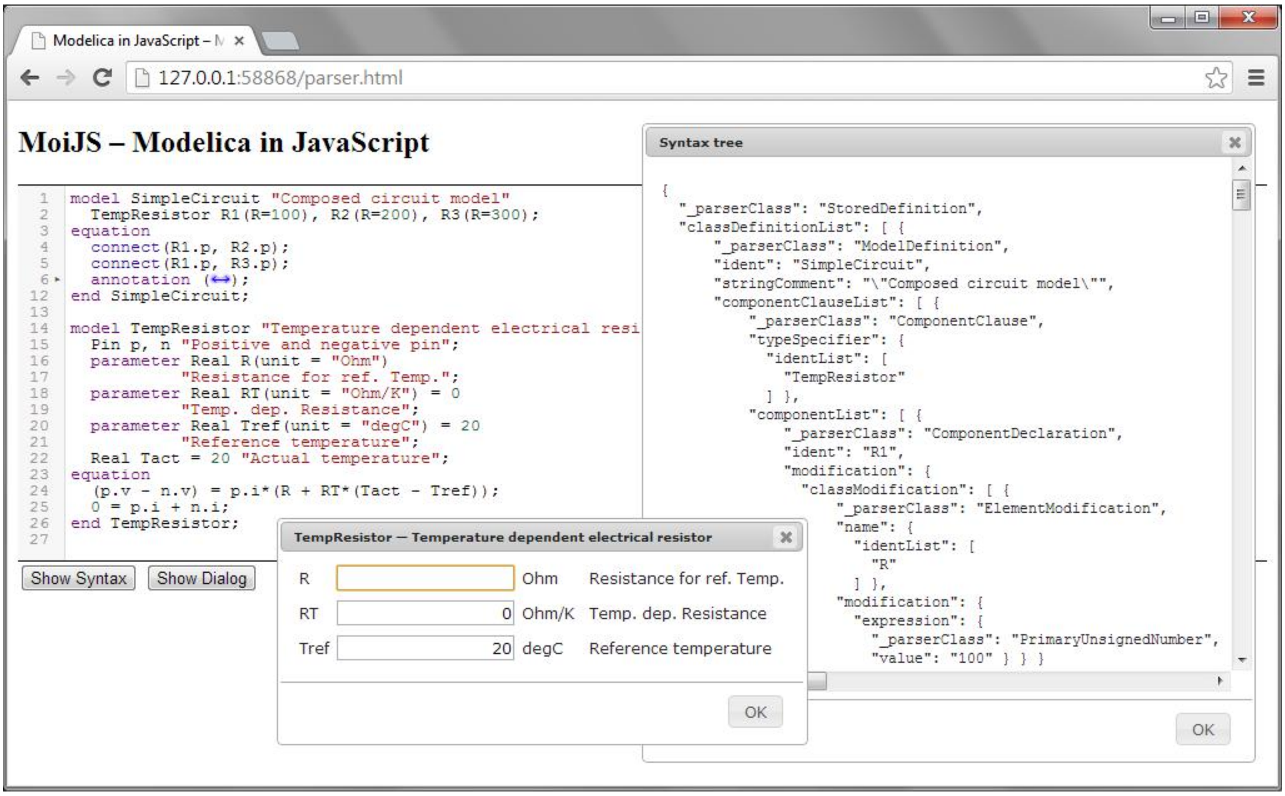
The Modelica parser in Python is extended with a backend translating algorithmic Modelica definitions to Python. This gives access to existing Python packages from scripted Modelica. It also enables the interactive debugging of algorithmic Modelica code.

The Modelica parser in JavaScript offers a generic backend interface. The paper demonstrates two applications. First a simple analysis tool for Modelica packages running from the command line is demonstrated. The true potential of JavaScript is the embedding of engineering data as Modelica code with HTML5 documents and their processing on the client side, e.g. in Web browsers. The paper shows a Modelica text editor and parameter GUI generator running in a web browser.

Client-side Modelica enables the use of the Modelica language itself as interface format in client/server architectures and for model exchange, instead of e.g. XML. Besides a significantly more compact syntax, this offers the advantage of having the semantics already standardized. The price to pay is a Modelica parser on the client side. This price turns out to be affordable in modern scripting or Web environments.

The availability of compatible JavaScript implementations by multiple vendors and fast just-in-time compilers being preinstalled on virtually any device make JavaScript attractive for more applications. Examples are HTML5 pages containing verbatim Modelica code that is evaluated on the client side, e.g. in a Web browser or in a modern mobile device.

MoiJS is extensible with new backends, exploiting the prototype based inheritance of JavaScript. MoiJS is available under the MIT license at http://omuses.github.io/moijs.



Dynamic modelling of a Condenser with the ThermoSysPro Library

Baligh El Hefni **Daniel Bouskela**
EDF R&D
6 quai Watier, 78401 Chatou Cedex, France
baligh.el-hefni@edf.fr daniel.bouskela@edf.fr

The condenser is a **two-phase** shell-and-tube heat exchanger. The feedwater flows inside the tube bundle, while the steam and condensate flows outside of those tubes located inside the cavity. In the condenser, there are two zones: the desuperheating zone and the condensation zone.

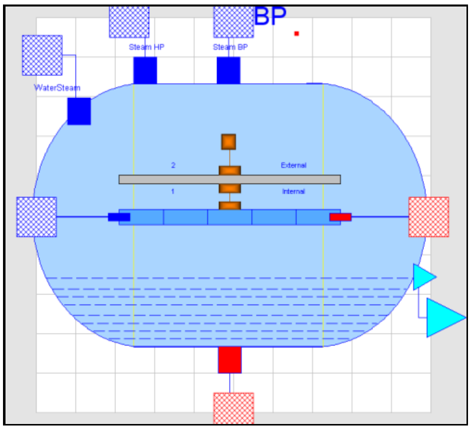
The condenser is an important device for the operation of power plants in particular for pressurized water reactors. Undesirable transients may lead to the automatic shutdown of the power plant.

To simulate the complex dynamic physical behaviour of the condenser, a dynamic model has been developed using Modelica. The component model is meant to be used for power plant modelling and simulation with the ThermoSysPro library developed by EDF and released under open source license.

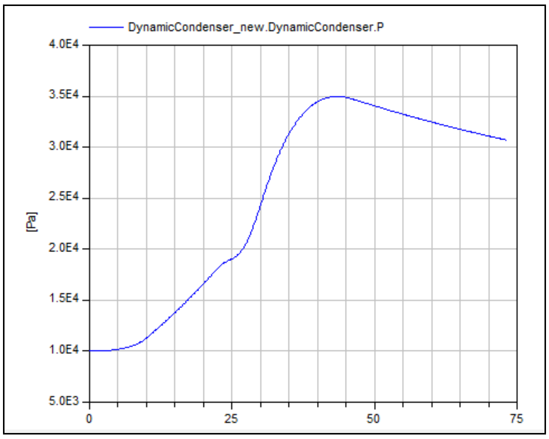
The transient most unfavorable to keeping the integrity of the condenser vacuum is the loss of the cold source pumps followed by a turbine trip and a missed islanding (house load operation).

The objective is to study the evolution of the pressure of the condenser for this kind of transients. During the transiernt, the condenser pressure should be always less than 5e4 Pa.

The present paper describes in detail the condenser model: hypothesis, governing equations, correlations and the test-case (structure of the test model, the parameterization data and the results of simulation).



Icon of the component model



Evolution of the pressure inside the condenser

References

[1] Collier J.G., and Thome J.R., 'Convective Boiling and Condensation', Mc Graw-Hill Book Company (UK) limited, 1972 Clarendon Press, Oxford, 1996.

[2] Patankar S.V., 'Numerical Heat Transfer and Fluid Flow', Hemisphere Publishing Corporation, Taylor & Francis, 1980.

Wavelet Library for Modelica

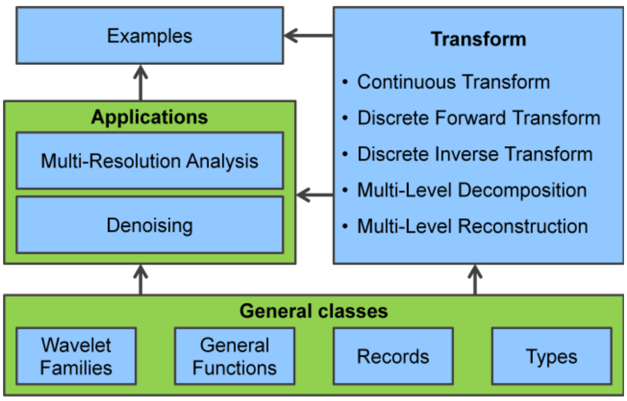
Jianbo Gao*, Yang Ji**, Johann Bals**, Ralph Kennel*
*Technische Universitaet Muenchen, Arcisstr. 21, 80333 Munich, Germany
michael.gao@tum.de, ralph.kennel@tum.de
** German Aerospace Center, Muenchner Str. 20, 82234 Wessling, Germany
yang.ji@dlr.de, johann.bals@dlr.de

A wavelet [1] library has been a standard component in many simulation programs. However, it has not yet been included in Modelica. So far some work has been reported to solve different problems combing Modelica simulation and wavelet calculation [2][3]. This showed the requirement of the wavelet analysis within Modelica. To fill the blank, a comprehensive wavelet library has been developed for Modelica. The preliminary version of this library had already been reported in 2012 and tested with a practical problem [4].

This library includes fifteen commonly used wavelet families. It can carry out continuous wavelet transform, forward and inverse discrete wavelet transforms, and multi-level decomposition and reconstruction in one-dimensional space.

In addition, special application tools for multi-resolution analysis and wavelet denoising are provided in this library, as well. Some examples are programmed to provide the users a quick start to build up their own algorithms. Nevertheless, the library is also well documented with three help files, a User's Guide, a Library References and a HTML Help Folder with hyper links.

This library was compiled and tested according to the Modelica language specification 3.2 under the Dymola platform version 2013. The test results prove its correction and functionalities.



Structure of the Wavelet Library

References

[1] I. Daubechies. Ten Lectures on Wavelets. SIAM 1992.

[2] T. Bunte, A. Sahin and N. Bajcinca. Inversion of Vehicle Steering Dynamics with Modelica / Dymola. Proceedings of the 4th International Modelica Conference, pp 319-328. Hamburg, Germany, March 7-8, 2005.

[3] Y. Ji and J. Bals. A Modelica signal analysis tool towards design of More Electric Aircraft. Proceedings of 2010 International Conference on Information and Applied Electronics, pp 152 - 156. Chendu, China, June 9-11 2010.

[4] J. Gao, Y. Ji, J. Bals and R. Kennel. Fault Detection of Power Electronic Circuit using Wavelet Analysis in Modelica. Prodeedings of the 9th International Modelica Conference, pp 513 – 521. Munich, Germany, Sept. 3-5, 2012.

Model-Based Energy Recuperation of Multi-Axis Machines

Marco Franke, Tamás Juhász, Ulrich Schmucker
Fraunhofer Institute for Factory Operation and Automation IFF
Sandtorstr. 22, 39106 Magdeburg, Germany
{marco.franke, tamas.juhasz, ulrich.schmucker}@iff.fraunhofer.de

This paper describes a new approach to sizing an electric drivetrain, including its power supply. The advantages of a real battery testing system are combined with the advantages of a Modelica model of a product. A testing system analyzes battery performance under specific constraints such as differing temperatures, vibrations and humidity. Since these constraints are hard to replicate in a model, an experimental rig is an optimal solution for battery tests [1]. Virtual engineering of a real system is advantageous in terms of the speed of modifications, the measurement of all relevant data and the low cost of development. The coupled virtual model and experimental testing rig for batteries constitute the basis for this new concept and improve the development of the electric powertrain.

References

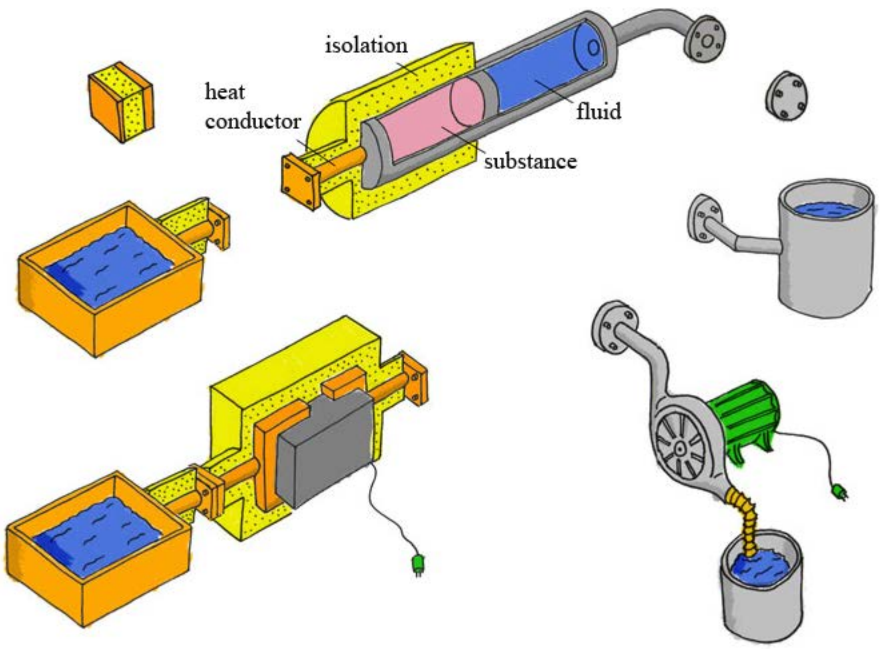
[1] Benecke, I.; Schmucker, U.: “Simulation statt Prüfstand” in Quality Engineering Vol 04.13, 2013, ISSN 1436-2457, pp. 40-42. (in German)

Systems Physics Library

Werner Maurer Elisabeth Dumont
ZHAW/IAMP
Technikumstrasse 9, CH-8401 Winterthur
maur@zhaw.ch dumo@zhaw.ch

In this poster the *Modelica Systems Physics library* is presented. The Systems Physics library is a free open-source library with models for different areas of physics. The primary use of the library is for educational purpose in Systems Physics.

The library contains models from five different domains (hydraulics, translational and rotational mechanics, electrodynamics and thermodynamics). Later on we will add chemistry as a sixth domain. Each domain contains connectors for one quantity and the corresponding potential, basic models (capacitance and resistance), sensors and actuators as well as some domain specific elements like nonlinear accumulator, nonlinear resistors, valves, springs or inductances. In addition to the constitutive equations each model contains an energy balance. Therefore dissipative elements calculate the loss of energy and some of them determine the entropy production with the help of an additional thermodynamic connector.



References

[1] http://code.google.com/p/system-physics/

[2] Borer, T et al.(2010). Physik – Ein systemdynamischer Zugang für die Sekundarstufe II. hep, Bern.

[3] Karnopp, D., Margolis, D., Rosenberg, R. (1990). System dynamics: a unified approach. Wiley, New York.

[4] Fuchs H (2010). The dynamics of heat. Springer, New York.

Implementation of the Omni Vehicle Dynamics on Modelica

Ivan Kosenko[1] Kirill Gerasimov[2]
[1]Moscow State Technical University of Radio Engineering, Electronics, and Automation
Department of Engineering Mechanics
Vernadsky Avenue 78, 119454, Moscow, Russia
kosenko@ccas.ru
[2]Lomonosov Moscow State University
Department of Theoretical Mechanics and Mechatronics
Leninskiye Gory 1, Main Building, GSP-1, 119991, Moscow, Russia
kiriger@gmail.com

Investigation of omni vehicle dynamical properties is sufficiently popular topic in frame of the multibody dynamics [1, 2, 3, 4]. Omni wheel is defined as one having rollers along its rim. Respectively omni vehicle is one equipped by omni wheels. Several steps of development of dynamical model for the omni vehicle multibody system are implemented.

The contact tracking using simplified and efficient algorithm turns out being possible for the roller. On the next stage the omni wheel model is developed and debugged. After that the whole vehicle model is assembled as a container class having arrays of objects as instantiated classes / models of omni wheels and joints. Dynamical properties of the resulting model are illustrated via numerical experiments.

Here we rely upon the “simple”3D multibody dynamics library classes utilized previously in several examples of the multibody systems dynamics [5]. Simultaneously this library enables us to create complex dynamical models including unilateral constraints of different nature.

Unlike to [2, 3] we emphasize here on the details of the unilateral constraint implementation paying special attention to contact switching when rollers changing.

References

[1] Campion, G.; Bastin, G.; d’Andréa-Novel, B.: Structural Properties and Classification of Kinematic and Dynamic Models of Wheeled Mobile Robots. IEEE Transactions on Robotics and Automation, Vol. 12, No. 1, pp. 47–62, 1996.

[2] Kálmán V.: Controlled Braking for Omnidirectional Wheels. International Journal of Control Science and Engineering, Vol. 3, No. 2, pp. 48–57, 2013.

[3] Tobolár, J.; Herrmann, F.; Bünte T.: Object-oriented modelling and control of vehicles with omni-directional wheels. Computational Mechanics 2009. Hrad Nectiny, Czech Republic, November 9–11, 2009.

[4] Zobova, A. A.; Tatarinov, Ya. V.: The Dynamics of an Omni-Mobile Vehicle. Journal of Applied Mathematics and Mechanics, Vol. 73, Iss. 1, pp. 8–15, 2009.

[5] Kosenko I. I., Loginova M. S., Obraztsov Ya. P., Stavrovskaya M. S., Multibody Systems Dynamics: Modelica Implementation and Bond Graph Representation. In: Proceedings of the 5th International Modelica Conference, arsenal research, Vienna, Austria, September 4–5, 2006, pp. 213–223.

Control and Characteristic Map Generation of Permanent Magnet Synchronous Machines and Induction Machines with Squirrel Cage

Marco Kessler
Modelon GmbH
Germany
Marco.Kessler@modelon.com

Markus Andres
Modelon GmbH
Germany
Markus.Andres@modelon.com

Thomas Schmitt
Modelon GmbH
Germany
Thomas.Schmitt@modelon.com

Due to different requirements for simulation during the design process of an electric drive system, investigations were carried out to supply models of different stages with consistent sets of parameters. Therefore physical models of the Modelica Standard Library (MSL) were equipped with state-of-the-art control strategies to operate in realistic conditions. These include field-oriented algorithms for permanent magnet and induction machines including field-weakening, current and voltage limitations as well as maximum torque per ampere blocks.

In the full paper it is presented how the losses computed by the physical model are transfered to the characteristic maps. Those losses are stored in tables to speed up simulation in cases where dynamics are of minor interest. These table-based models can then be used for energetic, thermal and life-time analysis with a consistent set of parameters generated from their physical counterparts. The physical models from which the map-based models were created can be applied when dynamics or other detailed effects in the machine or the controller are of interest.

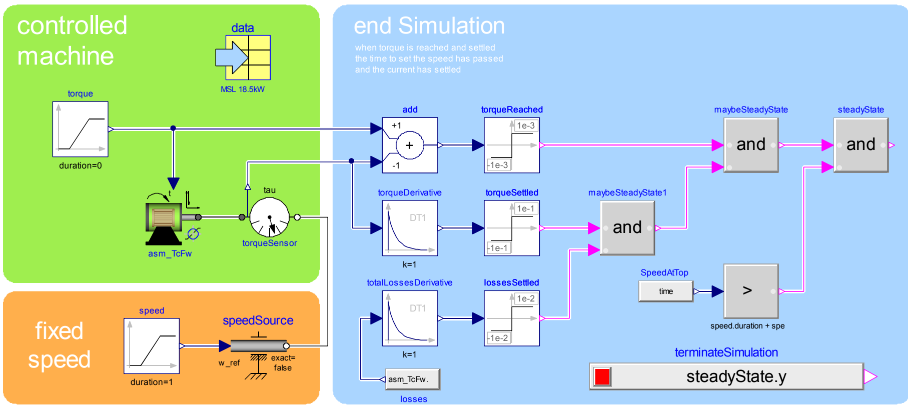


Figure 1: Model for loss calculation of a torque controlled induction machine

The transformation from physical models is done by a *Modelica* function that calls the model shown in Figure 1 within a loop to compute the loss table. This comes with the advantage, that a more precise machine model can simply replace the current model from the MSL with the advancements in modeling being directly represented in the characteristic map. Moreover the influences of the control algorithm like maximum torque per ampere are reflected in the characteristic map.

In the presented case the characteristic loss map was created depending on the angular velocity and the torque supplied by the machine. The results in the full paper show that errors smaller than 1% can be realized when doing energetic investigations. The speed-up in simulation times are found to be between a factor of 7 and ≥ 200. This factor is likely to rise if more complex models are applied.

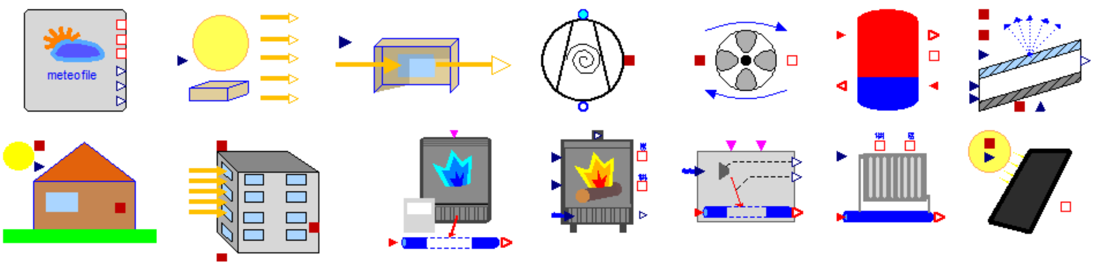
BuildSysPro: a Modelica library for modelling buildings and energy systems

Gilles Plessis Aurélie Kaemmerlen Amy Lindsay
EnerBaT – EDF R&D
Site des Renardières, 77818 Moret sur Loing CEDEX, FRANCE
gilles.plessis@edf.fr aurelie.kaemmerlen@edf.fr amy.lindsay@edf.fr

This paper presents the BuildSysPro Modelica library developed by the department of Energy in Buildings and Territories (EnerBaT) of EDF R&D.

Since the building sector is one of the main energy consumers nowadays, energy policies drive existent and new buildings towards better performances. These evolutions raise quantity of questions regarding their ability to ensure the occupants' health and comfort while decreasing energy consumption and increasing energy efficiency. These questions rely strongly on multi-domain representations including thermal, electrical, hydraulic or chemical processes. Modelica being an object-oriented, equation based language, is therefore well suited to represent this kind of coupled problems and complex systems.

The EnerBaT department of EDF R&D developed its own Modelica library, BuildSysPro, in order to perform multi-scale and multi-domain modelling. The choice of a new library was dictated by research needs, very specific for an energy producer and retailer, since they cover many domains. This library is designed to be used in several contexts including building physics research, global performance evaluation, technology development and impact assessment. It is also a basis for urban and building stock simulation. BuildSysPro is intended for a relatively large audience ranging from R&D scientists to building services engineers.



BuildSysPro contains classes to describe the whole building and its energy systems. These include boundary conditions models, envelope components, HVAC systems and other energy conversion devices (DHW, thermal and photovoltaic panels…) and their control systems. BuildSysPro provides a comprehensive set of elementary 0D/1D components which can be used for both static and dynamic applications. It is principally based on two branches of physics: pure thermal and thermo-fluid dynamics modelling. These classes are compliant with the *Thermal.HeatTransfer* and *Media* packages of the Modelica standard library to ensure a good level of interoperability with other Modelica libraries. BuildSysPro in its current version contains around 380 models and 130 functions. The BuildSysPro library has already been successfully used in published studies including:

- Technology performances and impact assessment,
- Sensitivity analysis regarding experimental validation,
- Urban simulation.

This paper presents the structure and some key elements of the library such as building envelope components, boundary conditions and HVAC systems. A focus is then made on validation through numerical comparisons with the IEA BESTEST procedure. Finally the use of BuildSysPro is described on a basic use case aiming at analysing the matching between heat demand and supply in a residential building. For more complex applications and validations, other papers are referred to and can be consulted.

Keywords: Modelica library; Building; Dynamic simulation; Numerical validations, Energy systems

Efficient Numerical Integration of Dynamical Systems based on Structural-Algebraic Regularization avoiding State Selection

Lena Scholz Andreas Steinbrecher
Technical University Berlin, Department of Mathematics
Str. des 17. Juni 136, 10623 Berlin
{lscholz, anst}@math.tu-berlin.de

Differential-algebraic equations (DAEs) naturally arise in the modeling of dynamical processes, in particular using MODELICA as modeling language. In general, the model equations can be of higher index, i.e., they can contain *hidden constraints*, which lead to instabilities and order reductions in the numerical integration. Therefore, a regularization or remodeling of the model equations is required. One way to obtain the required information on the hidden constraints is a structural analysis based on the sparsity pattern of the system. For the determination of a regular index-reduced system formulation then, usually, a crucial step is the *state selection* that is required in order to introduce new algebraic variables for the selected differential components of the DAE system. However, the choice of states that are selected can change during the numerical integration.

In this paper, we present a new regularization approach for the remodeling of quasi-linear DAEs that uses the information obtained from the structural analysis, in particular from the Signature Method [2], to construct a regularized overdetermined system formulation. The structural analysis provides the required information that allows to extract all hidden constraints. Adding these hidden constraints to the system leads to an overdetermined system formulation that has the great advantage that all constraints are stated in explicit form and no further analytical manipulations for the determination of a square and uniquely solvable system have to be applied. This overdetermined system can then be solved using a specially adapted numerical integrator that ensures that all constraints are satisfied during the numerical integration. Thus, the state selection is performed within the numerical integrator during runtime of the simulation. A further advantage of an overdetermined regularization is the possibility to add solution invariants, e.g., conservation laws, to the constraints, which often stabilizes the numerical integration.

Currently, no MODELICA simulation framework is able to handle overdetermined system formulations. Therefore, a prototype MODELICA framework `MPSSim` is presented that uses the translator `MO2FOR` [1] to translate an overdetermined system model provided in MODELICA into FORTRAN source code which can then be integrated using the software package `QUALIDAES` that is suited for the direct numerical integration of regularized overdetermined model equations. A comparison of `MPSSim` with simulation tools like `MapleSim`, `Dymola` and `OpenModelica` shows its promising performance.

References

[1] R. Altmeyer and A. Steinbrecher. Regularization and numerical simulation of dynamical systems modeled with Modelica. Preprint 29-2013, TU Berlin, 2013.

[2] J. Pryce. A simple structural analysis method for DAEs. *BIT Numerical Mathematics*, 41:364–394, 2001.

[3] L. Scholz and A. Steinbrecher. A combined structural-algebraic approach for the regularization of coupled systems of DAEs. Preprint 30-2013, TU Berlin, 2013.

Symbolic Initialization of Over-determined Higher-index Models

Lennart Ochel   Bernhard Bachmann
lennart.ochel@fh-bielefeld.de   bernhard.bachmann@fh-bielefeld.de
Bielefeld University of Applied Sciences, Department of Mathematics and Engineering
Am Stadtholz 24 - 33609 Bielefeld, Germany

Francesco Casella
francesco.casella@polimi.it
Politecnico di Milano, Dipartimento di Elettronica, Informazione e Bioingegneria
Piazza Leonardo da Vinci, 32 - 20133 Milano, Italy

The quantity of initial equations required in an object-oriented model can only be determined at system level. Since Modelica models are generally designed by components, it is difficult to calculate the amount of initial equations needed at system level, especially when changes are applied to the model, e.g. by adding or removing components. Therefore it is more convenient to define initial equations at component level. Consequently, the number of these equations is equal to the number of dynamic variables, i.e., the potential states. Due to component connections, algebraic dependencies between states may be introduced, which eventually lead to the removal of states when symbolic index reduction algorithms are applied. In this process, the corresponding initial equations are not automatically removed, which results in an over-determined initial system.

It is agreed that index reduction is necessary in object-oriented modeling to achieve full modularity without compromises, and suitable means to handle it have been developed over time, so it is obviously necessary to extend the handling to initialization as well. In the majority of cases the over-constrained initialization problem turns out to be consistent, and should therefore be handled automatically, without any intervention by the end user; inconsistent initialization problems should be reported in a user-friendly way.

This paper describes a symbolic algorithm that detects such redundant equations and determines if they are consistent or not. Consistent redundant initial equations can thus be removed automatically, and inconsistent ones can be reported to the modeler. The algorithm is implemented in OpenModelica, tested on several representative cases, and compared to previously presented concepts.

It works well for the recently developed package *OverdeterminedInitialization* from *ModelicaTest* containing various MSL-based test models, which end up in an recursively evaluable initial system. Furthermore, the developed algorithm takes care of singularities, if they occur during the consistency check. More complex problems end up with systems including algebraic loops. If they are not solvable symbolically, a numerical fall-back solution as well as advanced symbolic techniques are proposed.

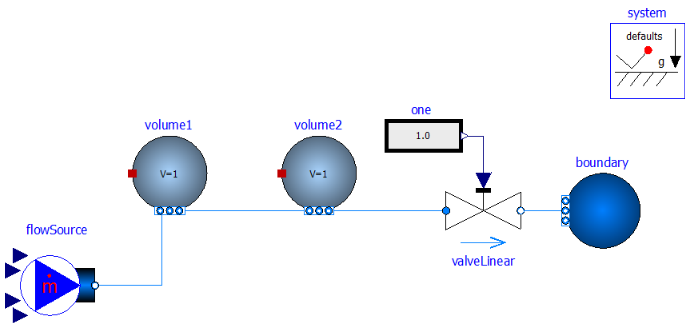
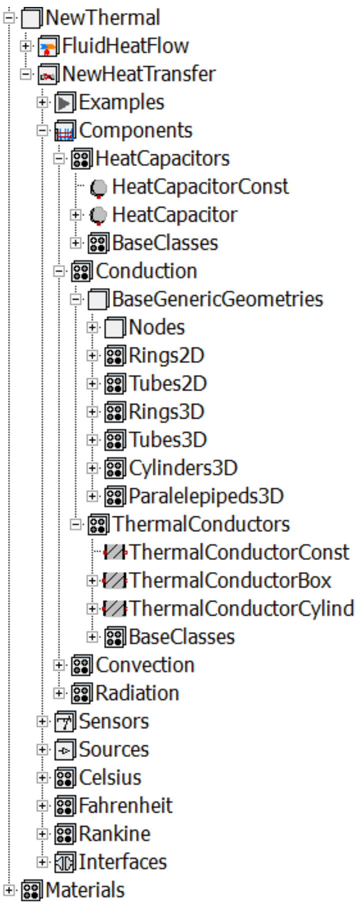


Figure 1: Over-determined fluid-example due to algebraic dependent states

Proposal for standardization of Heat Transfer Modelling in NewThermal Library

Susana López Itzal del Hoyo
IK4-TEKNIKER
Iñaki Goenaga, 5. 20600 Eibar, Gipuzkoa (Spain)
susana.lopez@tekniker.es itzal.delhoyo@tekniker.es



This article presents the `NewThermal` library that extends the capacities of `Thermal` library from the Modelica Standard Library (MSL) including a proposal for standardizing the use of `Material` models. The new library is intended to decouple the models that collect the equations of heat transfer phenomena from the thermo-physical properties of the matters (fluids and solids).

The `NewThermal` library, in the same way that the current `Thermal` library from MSL, is composed of thermal system components to model heat transfer and simple thermo-fluid pipe flow. Nevertheless, the models from the package proposed inherit the thermal properties from `Media` and `Material` models of the fluids and solids involved (either temperature dependent or constant). In this way, the user has three aspects to define; the heat transfer phenomena to be modelled, the geometrical characteristics of the bodies, and the matters involved.

Components inside `HeatTransfer` package are implemented such they can be used for any material model in `Materials` package, in the same way that components from `Modelica.Fluid` were carried out for their use with media models from `Modelica.Media`.

The `NewThermal` library, in addition, provides some general base models for the modelling of 2D and 3D heat conduction in basic solid geometries.

Two examples of use for different domains are presented to illustrate the features of the new libraries.

A Modelica Power System Component Library for Model Validation and Parameter Identification

Luigi Vanfretti[1,2] Tetiana Bogodorova[1] Maxime Baudette[1]
1:Smart Transmission Systems Lab. (SmarTS Lab), Electric Power Systems Department
KTH Royal Institute of Technology, Stockholm, Sweden.
2: R&D Division, Statnett SF, Oslo, Norway

Abstract

This paper summarizes the work performed in one of the work-package of the FP7 ***iTesla*** project. This work consisted in the development of a power system component library for phasor time domain simulation in Modelica.

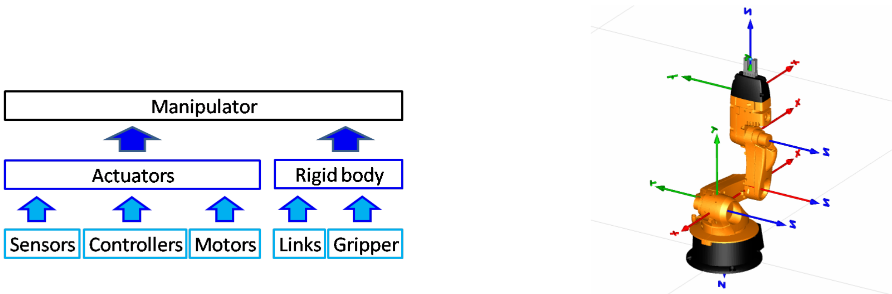
The models were used to build power system network models, used in experiments for parameter identification. The experiments were carried out with the RAPID toolbox, which has been developed at SmarTS Lab within the same project. The toolbox was written in MATLAB, making use of FMI Technologies for interacting with Modelica models.

Keywords: Power Systems, Phasor Simulation, Modelica, FMI, Parameter Identification, Model Validation.

Modelica Model for the youBot Manipulator

Rhama Dwiputra* Alexey Zakharov† Roustiam Chakirov* Erwin Prassler*
Bonn-Rhein-Sieg University of Applied Sciences, Department of Computer Science
Grantham-Allee 20, 53757 Sankt Augustin

Models and simulation tools are crucial in robotic research. Although there have been major improvements in the electronic and mechanical field, robots are still expensive equipments. The use of models and simulation tools overcome this problem. This paper presents the development of the Modelica model for the youBot manipulator. The youBot is a mobile manipulator designed to serve as the reference platform for industry, research and education [1]. Therefore, the Modelica model of the youBot manipulator is of high importance. The model was developed with a Modelica library for the manipulator's components which provides modularity, reusability and abstraction (Figure 1). This approach enables component exchange and component-based experiment of the developed model.



(a) Abstraction Layer (b) Visualization

Figure 1: The youBot's Manipulator Model

A model is a representation of the actual system and the benefit of having a model only holds true when the model accuracy is known. Simulation can result in wrong conclusion when the researcher forget the limitations and condition under which the simulation is valid [2]. Therefore, a comparison test with the actual system is performed to evaluate the model accuracy and identify the major components which require further improvement. The test result shows that the model reflects the actual system within a reasonable deviation. For future work, the manipulator model is planned to be to be tested with other Modelica tools (OpenModelica, jModelica) and used for hardware-in-the-loop experiments. The development or design of other manipulator models is also possible through the reusability of the components library. The library is publicly available[1] to be used for education and research.

References

[1] R. Bischoff, U. Huggenberger , and E. Prassler, *"Kuka youbot - a mobile manipulator for research and education"*, in *IEEE Int. Conf. on Robotics and Automat. (ICRA)*, pp. 1-4, May 2011.

[2] P. Fritzson, *Principles of Object-Oriented Modeling and Simulation With Modelica 2.1*. IEEE Press, 2004.

*rhama.dwiputra,roustiam.chakirov,erwin.prassler}@h-brs.de
†alexey.zhakarov@gmail.com
[1] www.youbot-store.com

Abstract of Equation based parallelization of Modelica models

Marcus Walther Volker Waurich Christian Schubert Dr.-Ing. Ines Gubsch
Dresden University of Technology
{marcus.walther, volker.waurich, christian.schubert, ines.gubsch}@tu-dresden.de

Modelica has become a widely used standard to describe physical simulation models. Compiling such a model into binary code can be performed by applications like Dymola, SimulationX or OpenModelica. However, all these tools only create a single thread simulation code out of standardized Modelica models, which does not allow for a speed-up with modern multi-core CPUs. This is due to the dependencies among the model equations which have to be considered in order to distribute the tasks amongst several threads. To gain a speed-up from this architecture, software programs have to be partitioned into several independent parts. A common representation of these parts is called a task graph or data dependency graph[1]. The authors of this article have developed a module for the OpenModelica Compiler (OMC), which creates, simplifies and schedules such task graphs. The tasks are created based on the BLT (block lower triangular)-structure[2], which is derived from the right hand side of the model equations. A noticeable speed-up for fluid models on modern six-core CPUs can be achieved, as shown in figure 1.

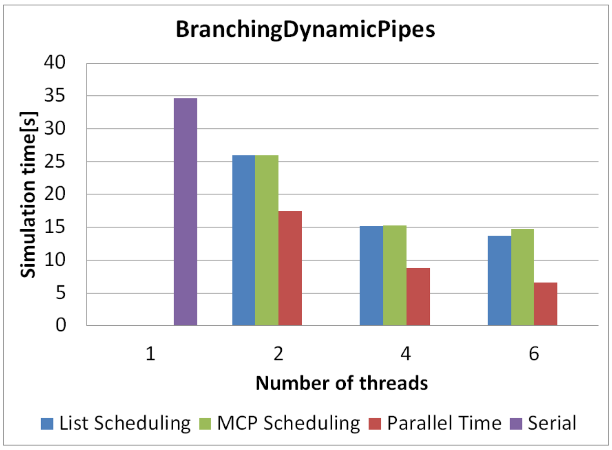


Figure 1: *Benchmark of the dynamic pipes example*

References

[1] P. Aronsson, *Automatic Parallelization of Equation-Based Simulation Programs*. Institutionen för datavetenskap, 2006.

[2] F. Casella, "A strategy for parallel simulation of declarative object-oriented models of generalized physical networks," *Linköping University Electronic Press, Linköpings universitet*, 2013.

Simulation of 2-dimensional flows in Modelica with the Casacaded Digital Lattice Boltzmann Method

Thomas Bäuml Helmut Kühnelt
AIT Austrian Institute of Technology GmbH
Mobility Department, Electric Drive Technologies, Vienna, Austria
thomas.baeuml@ait.ac.at helmut.kuehnelt@ait.ac.at

Calculating the dynamics of fluid flows is an important topic in the field of simulation. Common practice is to simulate complex scenarios by utilizing Computational Fluid Dynamics (CFD). Despite its capability of representing fluid flows in a very detailed way it has the drawback of compatibility. Coupling with other physical domain simulations is only possible by co-simulation. In this contribution a general methodology for modeling 2D fluid flows in Modelica is shown. Whereas there are contributions were the Navier-Stokes equations are solved by a finite volume method, this work deals with modeling them with a Lattice Boltzmann Method.

The Lattice Boltzmann method (LBM) is a relatively new simulation technique for fluid systems that has attracted interest as alternative to the discretization of the Navier-Stokes equations. LBM is a mesoscopic approach for modeling macroscopic fluid dynamics based on the Boltzmann kinetic equation which describes the statistical behavior of a non-equilibrium thermodynamic system. The fluid motion is based on the collective dynamics of fictitious particles on the nodes of a regular lattice. The dynamics of these particles is designed to obey the basic conservation laws ensuring hydrodynamic behavior in the continuum limit.

In this paper the theory behind LBM and the modeling approach is described. The implementation in Modelica is shown and some examples are given. Finally a convenient approach for setting up the computational domain is shown.

References

FORM-L: A MODELICA Extension for Properties Modelling Illustrated on a Practical Example

Thuy NGUYEN
EDF R&D
6, quai Watier, 78400 Chatou, FRANCE

Systems engineering methodologies for complex systems increasingly rely on, or could benefit from, modelling and simulation. For MODELICA to support activities such as functional validation of system requirements, design verification against requirements, testing, dysfunctional analyses and verification of operational procedures, the ITEA2 MODRIO project is developing extensions to the language. One of them concerns formal requirements and properties modelling, and is called FORM-L (FOrmal Requirements Modelling Language). This paper presents the main concepts underlying FORM-L, and illustrates them with examples taken from a MODRIO case study, the Backup Poser Supply (BPS) system.

Section 2 presents the main objectives assigned to FORM-L. Section 3 introduces briefly the BPS case study in oder to provide a background context for the examples given in the floowing sections. Section 4 presents how FORM-L considers *functions*, *constants* and *fixed* variables. Section 5 introduces the notions of *condition* and *event*. Section 6 presents the notions of *properties*, *requirements*, *assumptions* and *guards*. Section 7 presents the notion of *time locator*, continuous or discrete. Section 8 presents how FORM-L views *sets* and *arrays*. Lastly, Section 9 presents how *actions* are modelled in FORM-L.

Keywords: physical modelling; requirements modelling; systems engineering

References

[1] Schamai W., Buffoni L., Bouskela D., Fritzson P., ‘Automatic Model Composition using Bindings in Modelica’, Modelica 2014 conference proceedings, Lund, 2014.

[2] Bouskela D., El Hefni B., ‘A physical solution for solving the zero-flow singularity in static thermalhydraulics mixing models’, Modelica 2014 conference proceedings, Lund, 2014.

[3] Bouskela D., Jardin A., Nguyen T. 'Innovative Modelling Architecture for Design Verification against System requirements', , Modelica 2014 conference proceedings, Lund, 2014.

[4] Bouskela Daniel, MODRIO Toolset architecture

[5] Fritzson P., Principles of object-oriented modelling and simulation with MODELICA

[6] Harel David, Statecharts: a visual formalism for complex systems, In Science of Computer Programming, vol. 8, pp. 231-274, 1987

[7] Jardin A., EuroSysLib - Properties modelling

[8] Marcon F., EuroSysLib – Synthesis of needs within EDF, EDF technical report, 2009

[9] Nguyen Thuy, MODRIO - FORM-L, 2013

[10] Nguyen Thuy, MODRIO - Methodology, 2014

[11] Nguyen Thuy, MODRIO - The Backup Power Supply (BPS) case study, 2013

[12] Rapin Nicolas, ARTiMon: outil de monotoring de propriétés

[13] Schamai W., Model-based verification of dynamic system behavior against requirements

[14] Thomas E., Chastanet L., EuroSysLib - Dysfunctional: use cases and user requirements, Dassault Aviation technical report, DGT116083B, June 2009

[15] Vered Gafni, SPEEDS: Contract Specification Language (CSL)

Statecharts as a Means to Control Plant Models in LMS Imagine.Lab AMESim

Vincent Berthoux Sébastien Furic Loïc Wagner
{vincent.berthoux,sebastien.furic,loic.wagner}@lmsintl.com
LMS Imagine S.A.
7 place des Minimes
42300 Roanne, France

This article introduces a new feature of LMS Imagine.Lab AMESim that allows users to define plant model controllers. We start by reviewing some challenging aspects of hybrid state machine handling in asynchronous Modelica-based physical simulation environments. We then describe the implementation available in AMESim, focusing on user interaction and especially static error checking and reporting.

Integration of OpenModelica in Ptolemy II

Mana Mirzaei Lena Buffoni Peter Fritzson
Department of Computer and Information Science (IDA),
Linköping University, Division SE-581 83, Linköping, Sweden

Distributed, concurrent systems are becoming increasingly common. However, they are complex to develop. Therefore a large number of computational models and tools for modeling and developing such systems has emerged. The Ptolemy project aims to support such heterogeneous modeling in Ptolemy II[1], an open-source software framework for modeling, simulation and design of large concurrent real-time systems. This framework is a system- level design environment that provides the possibility of combining several variants of models of computation (MoCs) in one hierarchical heterogeneous model. Ptolemy II also supports an actor-oriented view of a system where the basic building blocks of a system are concurrent components called actors which communicate through messages sent via interconnected ports.

In this paper we present the integration process of OpenModelica into the Ptolemy II framework:

- In order to integrate OpenModelica into Ptolemy II, a dedicated computation domain with the corresponding director must be defined. Since Modelica is a language designed for continuous and discrete event modeling modeling of physical systems and variables described using DAEs, the continuous-time domain in Ptolemy II which models physical processes and supports mixtures of discrete and continuous behaviors is considered the most suitable for the Modelica language. Therefore the OpenModelica domain extends the continuos time domain already present in Ptolemy.

- To simulate Modelica models in Ptolemy II, it is necessary to invoke the OpenModelica Compiler (OMC). OMC provides a CORBA interface for remotely invoking the compiler from client applications. This interface is used to communicate with OMC from Ptolemy II.

- Moreover, OpenModelica offers a user-interactive and time synchronous simulation known as OpenModelica Interactive (OMI). OMI is part of the simulation runtime core and is used for interactive simulation of Modelica models in Ptolemy II.

Integrating OpenModelica into OMI enables the Ptolemy II users to simulate Modelica models with the OpenModelica Compiler from within the Ptolemy environment and interface them with components from other computation models.

References

[1] Christopher Brooks, Edward A. Lee, Xiaojun Liu, Stephen Neuendorffer, Yang Zhao,Haiyang Zheng, Heterogeneous Concurrent Modeling and Design in Java (Volume 1: Introduction to Ptolemy II) , April 1, 2008.

[2] Peter Fritzson. Principles of Object-Oriented Modeling and Simulation with Modelica 2.1, 940 pages, Wiley-IEEE Press, 2004.

On Extending JGrafchart with Support for FMI for Co-Simulation

Alfred Theorin   Charlotta Johnsson
Department of Automatic Control, Lund University, Lund, Sweden

Sequential Function Charts (SFC) is one of the IEC 61131-3 PLC standard languages for sequential, parallel, and general state-transition oriented automation applications. SFC is supported by most large industrial automation systems, for example 800xA by ABB, SIMANTIC S7 by Siemens, RSLogix 5000 by Rockwell Automation, DeltaV by Emerson, and CENTUM CS by Yokogawa. SFC is widely used and accepted for industrial automation, but is a low level programming language and thus implementing larger applications in SFC is inconvenient.

Grafchart has been developed for automation applications with focus on scalability. It extends SFC with high level features such as hierarchical structuring, reusable procedures, and exception handling. This makes it convenient to implement large applications that are overviewable and maintainable. Grafchart and SFC have the same graphical syntax with steps, possible application states, and transitions which change the application state.

JGrafchart [1] is a free integrated development environment for the Grafchart language. It is a research and education tool which is used for instance in the EU/GROWTH project CHEM for control in process industry, the EU FP7 project ROSETTA for robotic assembly, a master's thesis for modeling of avionics systems, and laboratory exercises on sequential and batch control.

JGrafchart can be connected to external environments through a multitude of customizable input/output (I/O) integration capabilities and can thus be used to control external real and simulated processes. As there is no built-in support for any particular simulation environment there is much potential for improvement in terms of effort for specifying the simulated model, quality of the models, support for inspecting simulation results, and time required to simulate, especially for more complicated physical systems.

Functional Mock-up Interface (FMI) is a standard which aims at combining dynamic models developed in various tools. A tool can export a model as a Functional Mock-up Unit (FMU) which can then be combined with other FMUs to compose the whole system. The FMI standard consists of two parts, namely *FMI for Model Exchange* and *FMI for Co-Simulation*. The difference is that for *FMI for Co-Simulation* a FMU also includes an individual solver to simulate its behavior. *FMI for Co-Simulation* enables simulation of coupled technical systems with focus on time-dependent problems.

Extending JGrafchart with built-in support for *FMI for Co-Simulation* gives more and better opportunities to connect JGrafchart to other simulation capable tools. It also makes a state machine related language with roots in industrial automation and with high level language features such as object orientation, hierarchical structuring, code reuse, and exception handling available for FMI co-simulation.

In this paper adding *FMI for Co-Simulation* support to JGrafchart is conceptually evaluated. It is discussed how JGrafchart fits into the *FMI for Co-Simulation* framework and potential ways to implement this are discussed.

References

[1] Lund University. *JGrafchart*. URL: http://www.control.lth.se/Research/tools/grafchart.html (visited on 2014-01-26).

Development of Custom Workflows for Simulation and Analysis of Functional Mock-up Units

Sureshkumar Chandrasekar[1]
[1] Modelon Inc.
Hartford, CT
United States
chandrasekar.sureshkumar@modelon.com

Jesse Gohl[2]
[2] Modelon Inc.
Ann Arbor, MI
United States
jesse.gohl@modelon.com

Abstract

Development of customized workflows and interfaces to deploy Modelica Functional Mockup Units (FMUs) with the various FMU tools has been gaining traction in the industry – both with tool vendors as well as end users.

The FMI Add-in for Excel (FMIE) is a commercial product from Modelon AB that enables the deployment of FMUs in Microsoft Excel. FMIE enables the user to programmatically control the add-in through Visual Basic code in Excel. This allows the implementation of custom workflows and interfaces for the user to interact and automate the tasks involved in loading, simulation and analysis of FMUs.

In this paper we present a workflow in FMIE for an automobile thermal management model FMU. The workflow utilizes Visual Basic scripts for automation and user-forms for user interaction.

Keywords: FMU;Automated workflow;FMI Add-In for Excel; Visualization; Design of Experiments, Batch Simulations, Monte-Carlo Analysis

References

[1] Modelisar ITEA2, "Functional Mock-up Interface for Model Exchange", Accessed from *www.fmi-standard.org* in Nov 2013.

[2] Modelisar ITEA2, "Functional Mock-up Interface for Co-Simulation", Accessed from *www.fmi-standard.org* in Nov 2013.

[3] Modelon AB, "FMI Add-In for Excel", Accessed from *http://www.modelon.com/products/fmi-add-in-for-excel/* in Nov 2013.

[4] Batteh J., Gohl, J., S Chandrasekar, "Integrated Vehicle Thermal Management in Modelica: Overview and Applications" *Proceedings of the 10th International Modelica Conference*, Lund, 2014.

A Medium Model for the Refrigerant Propane for Fast and Accurate Dynamic Simulations

Roozbeh Sangi Pooyan Jahangiri Freerk Klasing
Rita Streblow Dirk Müller
Institute for Energy Efficient Buildings and Indoor Climate,
E.ON Energy Research Center, RWTH Aachen University,
Mathieustr.10, 52074 Aachen, Germany
rsangi@eonerc.rwth-aachen.de

To address the main challenges arising from the ever-increasing energy demand worldwide and its associated environmental impacts, it is not only essential to optimize the existing energy landscape but it is also necessary to develop new approaches. One of these approaches is to effectively utilize low-exergy demand heating and cooling systems like conventional geothermal heat pump systems by expanding the usage of refrigerants from a simple refrigerant cycle to more complete systems such as direct exchange heat pump systems.

Investigating the use of different fluids and their advantages in new energy systems has increased a need for faster and more robust simulation models. There are different methods of simulating different refrigerants [1, 2] such as R600a, CO_2 and R134a. Although these methods can calculate the properties of these refrigerants accurately, they require a long simulation time. This problem motivated the authors of this paper to develop a fast, accurate and stable refrigerant model for the dynamic simulation of complete systems over a long period of time.

In this paper, developing and improving a medium model for propane is discussed. Besides being fast and accurate, the propane model should also be stable in different dynamic simulation scenarios.

First, one of the existing libraries, namely HelmholzMedia library is tested and in some cases modified to increase the stability. Since the simulation speeds were not in an acceptable range in the existing models, a new propane model based on the Helmholz Equation of States is introduced. Unlike the propane model of the HelmholzMedia library, this model is only valid in a certain range of pressures and temperatures. The range is chosen so that it includes a wide variety of engineering applications, corresponding to temperatures between -10°C and 70°C and pressure between 0.5 up to 30 bars. The new model is then tested as a refrigerant in a direct exchange heat pump system. A comparison between an existing propane model and the new model shows that much faster simulations, up to 30 times, are achievable with the new propane model without sacrificing the accuracy.

References

[1] C. Heinrich, K. Berthold "A Modelica Library for Simulation of Household Refrigeration Appliances Features and Experiences". In: *Proceedings of the 5th International Modelica Conference,* 2006, pp. 677-684.

[2] I. Bell, S. Quoilin, J. Wronski, V. Lemort "CoolProp: An Open-Source Reference-Quality Thermophysical Property Library" In: *2nd ASME ORC International Seminar*, 2013.

Consistent Simulation Environment with FMI based Tool Chain

Edo Drenth[1], Mikael Törmänen[2], Krister Johansson[2],
Bengt-Arne Andersson[1],Daniel Andersson[1], Ivar Torstensson[1], Johan Åkesson[1]

[1]Modelon AB
Ideon Science Park, Lund, Sweden
info@modelon.com

[2]Volvo Car Corporation
Dept. Complete Powertrain
Göteborg, Sweden

Abstract

Systems engineers face the ever increasing chase for reduced time to market, while the systems to develop ever increase in complexity. Software systems design and integration processes have therefor evolved along the well-known V-cycle.

This paper will focus on the software integration for mechatronic systems as they develop fast due to high demands and challenging requirements in the automotive industry.

The development order of model in the loop (MIL), software in the loop (SIL), processor in the loop (PIL) and hardware in the loop (HIL) can be seen as state of the art practised by many systems engineers. Driver in the loop (DIL) may be in its infancy, but rapidly growing.

The novelty presented in this paper is the consistency of the plant models used in the integration chain supporting consistent model data propagation: Functional Mock-up Units (FMU) defined by the open standard of the Functional Mock-up Interface[1] (FMI).

Keywords: FMI, FMU, MIL, SIL, PIL, HIL, plant models, Modelica

References

1. Blochwitz, T., Otter, M., Arnold, M., Bausch, C., Clauß, C., Elmqvist, H., Junghanns, A., Mauss, J., Monteiro M., Neidhold, T., Neumerkel, D., Olsson, H., Peetz, J.-V., Wolf, S., *The Functional Mockup Interface for Tool independent Exchange of Simulation Models*, 8th Modelica Conference, Dresden, Germany, 2011
2. Liebezeit, Bräuer, Serway, Junghanns: *Virtual ECUs for developing automotive transmission software*, 10th CTI Symposium Innovative Fahrzeug-Getriebe Hybrid- und Elektro-Antriebe, 5.-8.12.2011, Berlin, Germany
3. Junghanns, A., *Virtual integration of Automotive Hard- and Software with Silver*, Qtronic GmbH, Berlin
4. Andreasson, J., Andersson, D., Batteh, J., Gohl, J., Griffin, J., Krueger, I., *Integrated simulation of a e4WD vehicle using Modelica*, Advances in Automotive Control, Volume#7, Part#1, 2013
5. *FMI Toolbox User's Guide 1.7*, Modelon AB, Lund, Sweden, 2013
6. *Functional Mock-up Interface for Model Exchange, Version 1.0*
7. *Functional Mock-up Interface for Co-Simulation, Version 1.0*
8. https://www.fmi-standard.org/
9. *Measurement, ECU Calibration, and Diagnostics –Development Solutions for Automotive Embedded Systems*, ETAS GmbH, Stuttgart, Germany, 2010

A MATLAB to Modelica Translator

Mohammad Jahanzeb[1], Arunkumar Palanisamy[1], Martin Sjölund[1], Peter Fritzson[1]

[1]PELAB – Programming Environment Laboratory
[1]Department of Computer and Information Science
Linköping University, SE-581 83 Linköping, Sweden
mjahanzeb@live.com, {arunkumar.palanisamy, martin.sjolund, peter.fritzson}@liu.se

Matlab is a proprietary, interactive, dynamically-typed language for technical computing. It is widely used for prototyping algorithms and applications of scientific computations. Since it is a dynamically typed language, the execution of programs has to be analyzed and interpreted which results in lower computational performance. In order to increase the performance and integrate with Modelica applications it is useful to be able to translate Matlab programs to statically typed Modelica programs.

This paper presents the design and implementation of Matlab to Modelica translator. The Lexical and Syntax analysis is done with the help of the OMCCp (OpenModelica Compiler Compiler parser generator) tool which generates the Matlab AST, which is later used by the translator for generating readable and reusable Modelica code.

Keywords: Modelica, MetaModelica, Matlab, OMCCp, translation.

Setting up a framework for model predictive control with moving horizon state estimation using JModelica

Mats Vande Cavey, Roel De Coninck, Lieve Helsen,
KU Leuven, Department of Mechanical Engineering, 3001 Leuven
mats.vandecavey@mech.kuleuven.be, roel.deconinck@mech.kuleuven.be, lieve.helsen@mech.kuleuven.be

A framework is proposed that allows using Modelica models in a model predictive control framework with state estimation. The aim of the framework is to optimally control the heating of a building. The contribution lays in the implementation of the moving horizon estimation (MHE) using JModelica [FIXME ref] and Modelica models.

This framework is applied to heating control of a residential, single zone building. The controller model is a low order resistance-capacitance (RC) building model. The emulator is a 'detailed' Modelica model of the building and heating system. MHE is a state estimator which determines the (controller) model error for each state over a past time window. This is done by optimizing the trajectories of every state's model error over that past time window, to fit the measurements of one (or more) state(s). To this end, instead of the typical deterministic Modelica equations, the controller model needs to be formulated using stochastic equations for the states and ***w*** is formulated as an input to the Modelica model:

$$C * \frac{d(T)}{dt} = Q_{flow} + \boldsymbol{C} * \boldsymbol{w} \qquad (1)$$

$$\frac{\boldsymbol{d(\Delta T)}}{\boldsymbol{dt}} = \boldsymbol{w} \qquad (2)$$

The optimization problem is formulated as a least squares problem (3) with a different weighting for the terms:

$$\min_{x0,\{w_k\}_{k=T-N}^{T-1}} \sum_{k=T-N}^{T-1} \|v_k\|^2_{R^{-1}} + \|\Delta T_k\|^2_{Q^{-1}} \qquad (3)$$

The weighting factors $R^{-1}$and Q^{-1} denote the inverse of the covariance matrices of the measurement noise v_k and the model error (process noise) ΔT_k. If the covariance matrices are unknown, $R^{-1}$and Q^{-1} can be tuned in order to produce good results.

The input variable w is the result of the state estimation. Multiplied by the capacity C, the term represents an extra heatflux. It is linked to the model temperature through the state equation (1) and the end temperature difference ΔT of the MHE decides the new start temperature of the OCP. In Figure 1 the improvement of applying state estimation is shown, as the real zone temperature is kept closer to the optimal one.

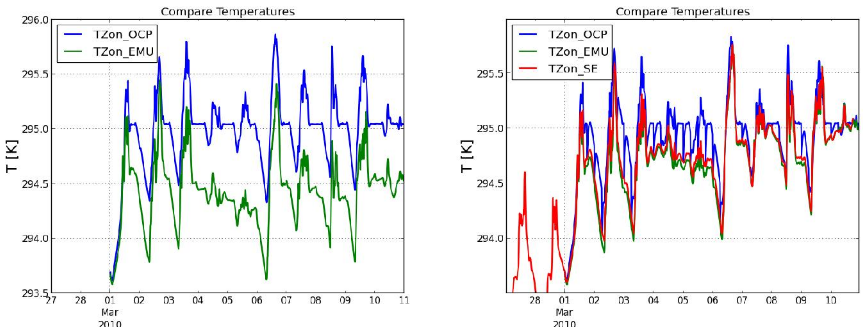


Figure 1: Comparing zone temperatures without (left) and with (right) state estimation