NordDesign 2012
August 22 – 24, 2012
Aalborg, Denmark

Information transfer from electrical design to simulation models in Modelica for virtual commissioning

Prof. Dr.-Ing. Martin Eigner[1], Dipl.-Ing. Radoslav Zafirov[1] and Dr.-Ing. Thomas Baudisch[2]

[1]Institute for Virtual Product Engineering
University of Kaiserslautern
Gottlieb-Daimler-Straße 44
Kaiserslautern, Germany D-67653
E-mails: eigner@mv.uni-kl.de; radoslav.zafirov@mv.uni-kl.de

[2]Siemens AG
Corporate Research and Technologies
Otto-Hahn-Ring 6
Munich, Germany D-81739
E-mail: thomas.baudisch@siemens.com

Abstract
With the industry focusing on adaptive production, the complexity of production processes and production systems is constantly growing. Therefore, efficient interdisciplinary modeling and validation within the engineering process of manufacturing plants is now more relevant than ever. One computer assisted validation method of increasing importance for plant engineering is the virtual commissioning. Virtual commissioning is a method to commission and functionally test the control code running on the real control hardware against a model of the plant. This validation approach can be time-saving, since commissioning and test of the controller software does not have to be performed on the spot, after the plant has been physically built and set up. On the other hand the setup of the virtual commissioning model is a manual process today that involves a lot of time. The model setup requires results from the preceding mechanical and electrical design as input. The generation of these simulation models is done in designated simulation tools.
Following the model based design paradigm, an approach has been developed, on how to automatically generate parts of multidisciplinary plant simulation models by re-using information from electrical design models. The goal of the approach is to avoid redundant manual re-modeling of aspects of the plant that have already been defined in electrical design.
The considered simulation models are discrete behavioral models created using the Modelica language. A prototype application is presented in the paper, which has been programmed to validate the approach of translating electrical design models to a sub model of the virtual commissioning overall model using real plant data. A discussion on the applicability of this approach in real engineering scenarios concludes the paper.

Keywords: mechatronic design, automatic model generation, virtual commissioning, Modelica.

Introduction

Interdisciplinary simulation addresses key problems in the design of mechatronic systems. It helps ensure design requirements are met in the final system and helps predict a product‘s behavior before physical prototypes exist. These two are among the top challenges of mechatronic product development whereby integration of electronics and software remains to be one of the main pressures that are driving companies to improve their mechatronic product development processes [1]. In the field of automated systems in materials handling, logistics and production there are existing approaches for simulation that allow for integrating discipline-specific design results to test aspects of a system‘s overall functionality [2]. Mechatronic simulation like virtual commissioning can be very beneficial especially when it comes to integrating controller software by testing it with simulation models of the plant thus guaranteeing functionality and faster ramp-up during the physical commissioning [3].

Creating the models for such interdisciplinary simulation involves nowadays immense extra effort. Whether it is modeling geometry and kinematics of assemblies for the Multi Body analysis and visualization or generating the behavior models of electrical schematics, it is not possible to further use results of discipline-specific design for interdisciplinary simulation with most tools currently on the market [4], [5]. Instead, existing information from discipline-specific models or documentation is being manually remodeled in the simulation domain.

This paper demonstrates an approach that can save modeling effort by further using information from electrical design to partially automate the generation of those sections of the interdisciplinary simulation model that refer to electrical design.

State of the art and related work

This section clarifies the state of the art and research prior to introducing the main approach. The content of models for virtual commissioning is discussed, in order to show the scope of this type of simulation. The multiphysics simulation language Modelica is introduced, noting its applicability for mechatronic simulation and showing its potential for easy integration with electrical design models. This is followed by the state of the art in electrical design, describing the possibilities for further use of electrical design model data in simulation models. The section concludes with a summary focusing on standards and approaches for further use of design information in the field of automation, which is the main goal of model based design.

Models for virtual commissioning as an interdisciplinary simulation

Based on model theory, a model only covers some of its original’s properties and is created for a purpose [6] [7]. In other words, a model is an abstract description of a system within an observed scope of application. The scope of application of virtual commissioning is to detect flaws in software code and other aspects of design in a real-time simulation and thus to prevent mistakes during construction and accelerate commissioning and ramp up [8].

The abstraction level of these interdisciplinary models is generally higher than the one in discipline-specific simulation. Geometry is used mainly for visualization of movement sequences and models from mechanical design are simplified to tessellated geometry for faster calculation. Real-time collision detection and physical simulation is also within the scope of application [9]. Simulation of electrical design in the context of virtual commissioning is for the most part time-discrete signal-based behavior simulation using library components corresponding to electrical components [10]. It often does not reach the detail level of discipline-specific electrical schematic simulation involving e.g. voltage, magnetic field progressions, temperature analysis on electrical equipment, Fourier-analysis [11] etc.

Models for virtual commissioning involve multiple engineering disciplines but have the needed level of abstraction suited to their purpose. Hence, it is a limited share of information from electrical design that needs to be transferred to models for virtual commissioning.

Multiphysics simulation language Modelica

Modelica is a non-proprietary modeling and simulation language, trademark of the Modelica Association. It can be used to solve a variety of problems that can be expressed with differential-algebraic equations describing the behavior of continuous variables. [12]

Modelica is built on the object-oriented paradigm that allows for creating physically relevant model components that can be hierarchically structured and reused [13]. Reusable model components can be stored in a library. A standard library is provided by the Modelica Association, which contains basic model components from many different disciplines, ranging from Multi-Body mechanical, through electrical analog and digital model components, to mathematical functions, thermal system components and state machines for modeling discrete behavior of systems. Modelica has the capability to bring these disciplines together based on the notion of connectors [13]: any two model components can exchange a variable through a connection between them as long as it is of the same type. This so-called multi-domain capability makes Modelica suitable for certain scenarios for virtual commissioning since it supports visualization with 3D-geometry, signal-based simulation of electrical schematics and signal exchange with a real or modeled controller.

Modelica is a textual language but the capability to reuse code from a library and the concept of connectors between model components also allows for modeling in a so-called schematic view. In this view instantiated model components from the library are represented as blocks and the connections between them – as connections in a connection diagram, very similar to an electrical schematic. Modelica also supports a non-interactive model animation view, where a Multi-Body model setup or simulation results are shown in their 3D-geometrical representation.

Models in Modelica do not have to be statement-driven, where an output variable is calculated from input variables and there is a pre-set direction of calculations. When modeling in this way, models have to be re-arranged or new models have to be created, depending on which variables in a mathematical equation are known and which need to be calculated. Modelica supports declarative, non-casual object-oriented modeling, allowing for defining and using model components that resemble their physical counterparts instead of signal blocks [13]. Equations are declared within every model component and the direction of the calculation depends on which variables are known and which are not in the particular test case. This makes it possible to connect model components without consideration of the signal flow during simulations.

For the current area of application this means that model components can be defined for single pieces of electrical equipment that can be later on connected to each other, similar to the way they are connected in the physical world. The concept of connectors and the non-casual modeling thus allow for simulation models in Modelica to resemble the system topology that is also present in discipline specific design, consisting of components and the connections between those components.

Modern Computer Aided Electrical Design (E-CAD)

The central means of expression and communication in electrical design are the electrical schematics. Unlike design documentation in other disciplines like mechanical design, electrical schematics focus on a function-oriented description of the system, where components and the connections between them (system topology) play a primary role compared to physical size, shape and positioning of electrical equipment within the system [14].

Computer Aided Electrical Design has initially rationalized the tedious process of manually drawing electrical schematics for example by managing the standardized electrical component symbols in libraries for convenient reuse. Nowadays E-CAD tools are not simply used to draw electrical schematics but to create semantically rich models that can be used as a basis for further rationalization of the design process. Deploying the object-oriented paradigm in E-

CAD has led to having a central model in the project, where every instantiated library object is stored only once, containing its different representations in electrical schematics as different views thus guaranteeing consistency in e.g. naming and hyperlinking between document pages [14]. With a central model and specific rules in place, E-CAD tools are capable of consistency-checking and reworking electrical schematics, also automatically generating whole lists and bills of material.

The use of semantically rich models and not only documents in electrical design shows a potential for effective further use of design information in the simulation domain. As already stated, only a limited share of information from electrical design is relevant for virtual commissioning. For an overview over the content of electrical design models in plant design, a breakdown of the general types of electrical schematics is given in the following.

Table 1. Information content in the general types of electrical schematics

Documentation type	Topology information	Functional information	Physical information
Single-line diagram	◐	◐	○
Circuit diagram	●	●	○
Wiring diagram	●	○	◐
Layout diagram	○	○	◐
Bill of material	○	○	◐

Single-line diagrams are illustrative circuit diagrams in a simplified form that provide an overview over the functionality of a circuit and the basic architecture of the designed plant [14]. Circuit diagrams on the other hand contain the complete information on connections between electrical components. These diagrams do not contain physical information on positioning, size and shape of electrical components (Table 1). Wiring, layout diagrams and bills of material on the other hand are types of documentation that do not focus on the functional description of the system but on information that is needed to physically build and deploy the plant [14]. While wiring diagrams with terminal schemes contain the complete information on wiring (with wire coloring etc.) between components, the function of the system cannot be easily recognized.

These different types of documents show the relations between electrical components (model objects) within an electrical design model from different perspectives. The model of a plant also contains information (descriptive metadata) on every single object, like: technical, functional, purchasing, mechanical etc. Currently some E-CAD tools even support the management of IEC 61131-3 function blocks attached to their respective electrical components within the library [15], which are pieces of controller software code [16]. This can accelerate controller programming immensely by allowing for considerable amount of the controller software to be configured automatically within E-CAD based only on the model topology. Managing behavior models of electrical components in simulation language code (e.g. Modelica, Simit) in E-CAD libraries could in a similar way accelerate the generation of simulation models for virtual commissioning but is not state of the art.

Model-based design and model transformation in the automation domain

Model-based systems engineering (MBSE) is the formalized application of modeling to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases. MBSE is part of a long-term trend towards model-centric approaches adopted by other engineering disciplines, including mechanical, electrical and software [17]. A benefit from adopting these model-centric approaches is a more convenient information exchange between disciplines based on data format standards between tools and model transformation, which can save effort and minimize human mistakes by automating the generation and modification of models where possible. The lack of interoperability between tools is in fact one of the reasons for the high interest to deploy MBSE in engineering [17].

There are many existing standards in the automation domain, covering different disciplines, e.g. formalized process descriptions for process planning [18] or the standardized programming languages for PLC software engineering [16]. There are also approaches on cross-disciplinary data and model interchange, like the XML-based format AutomationML („Engineering data exchange format for use in industrial automation systems engineering“) which is on the path of international standardization [19], yet far from wide spread use.

Model transformation, being initially an approach from Model-Driven Engineering within the Software Engineering domain, is making its first steps within the industrial automation domain. Initial research shows the applicability of model transformation in the conceptual design phase, where function-oriented models can be translated to first circuit diagrams and even controller source code [20]. In a similar manner, model transformation can be utilized for automated generation of simulation models on the component level for virtual commissioning further along the design process.

A method for using system topology information from E-CAD models to generate models in Modelica

As explained so far, there are many similarities in modeling in electrical design and modeling for interdisciplinary simulation and potential exists, on how to transfer information from one to another to guarantee less modeling effort and higher model consistency. With both modeling approaches relying on the object-oriented paradigm, there are many similarities at the library level (Figure 1). As previously stated, the non-casual modeling in Modelica allows for defining classes and thus model components in the library that refer to real world physical components.

Electrical Design tools	Simulation language Modelica

Component library E-CAD:
• *device type name*
• *device connectors*
• *device symbols (standardized)*
• *2D-geometry*
• *device connectivity rules*
• technical metadata
• manufacturer metadata
• purchasing metadata
• …
	Component library Modelica:
	• *component type name*
	• *component connectors*
	• *diagram representation (not standardized)*
	• *2D/3D-geometry*
	• *model connectivity rules*
	• behavior description/model code:
	  • model parameters, variables, constants
	  • model equations
	• …

E-CAD model instances: • *device tags* • *connections (pneumatic/hydraulic/electric)* • electrical schematics page numbering • device wiring information • plant layouts • …	Behavior model instances: • *model component instance names* • *connections (pneumatic/hydraulic/electric)* • simulation types • simulation parameters • simulation result presets • …

Examples for objects in both libraries:
• clamps
• valves
• optical sensors
• transformers

Figure 1. Similarities in E-CAD and Modelica at library and model instance level (italic)

Thus, in the application area of modeling the behavior of electrical devices for virtual commissioning, library components in Modelica can resemble specific electrical devices. Similar to their representation in electrical design, they possess connectors to be interlinked with other model components in plant model instances. The number of connectors does not necessarily match due to simulation model components being used in a multidisciplinary context. An electric drive may have, for example a mechanical connector (torque) in addiction to electrical connectors.

Similarities in the information contained in E-CAD and in Modelica cover the system's topology for the most part (Figure 1, italic). Other information is discipline-specific. The models generated with this approach are built on three levels, considering the specifics of the Modelica language and the specifics of electrical design:

Model component level

Some descriptive metadata on electrical devices in E-CAD and the definition of device behavior in Modelica code are examples for information that is specific to electrical design and to simulation (Figure 1, non-italic, library level). Such information does not get transferred between Modelica models and E-CAD models and is managed within tool libraries.

Plant model level

Information referring to the system's topology, like device instance names and device connections (Figure 1, model instance level) is transferred from electrical design to the simulation model in Modelica to save modeling effort and guarantee model consistency.

Model hierarchization level

Even the simplest automated plants consist of many single devices, the use of which is shown in electrical schematics. To maintain an overview over the whole plant, circuit diagrams are organized in pages with numbered hyperlinks from one page to another. Modelica on the other hand also supports hierarchization by including model components to higher level compounds and forwarding free connectors to these compounds (Figure 2).

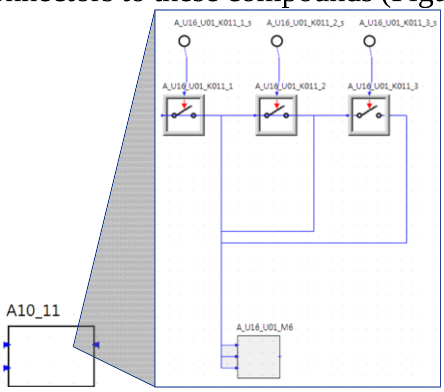

Figure 2. Model hierarchization in Modelica

This construct was used to organize the simulation models in Modelica, since electrical schematics often come in hundreds of pages. The most appropriate type of arrangement of model components within the compounds is the arrangement of devices on circuit diagrams in E-CAD since this documentation type contains the whole model topology and gives the best functional overview on the system (Table 1) which is also nice to have in simulation.

Demonstration with an example

The method of transferring the system topology from E-CAD models to create simulation models in Modelica was validated with a demonstrator application. The electrical design of a conveyor belt was used as an example. It involves primarily electrical drives with relay logic and signal in- and outputs of the Programmable Logic Controller (PLC).

The tool used in electrical design was COMOS 9.1. As a state-of-the-art tool it deploys the object-oriented approach and can be used to create semantically rich models that can be exported to XML. The tool used for modeling and simulation with Modelica was SimulationX 3.4 which supports the Modelica specification.

The implementation of the method was tool-specific. XML files were used to export the plant topology information and device positioning on circuit diagrams in COMOS. Two ways of automatic model generation were possible with SimulationX. The first involves a translation in Modelica code that references to the library components that need to be instantiated and interconnected. The second involves a translation into step-by-step commands for building the model that can be saved in an executable script file. The second approach yielded better results with the tool SimulationX and was therefore used for the implementation of the demonstrator application “COMOS to SimulationX model converter”.

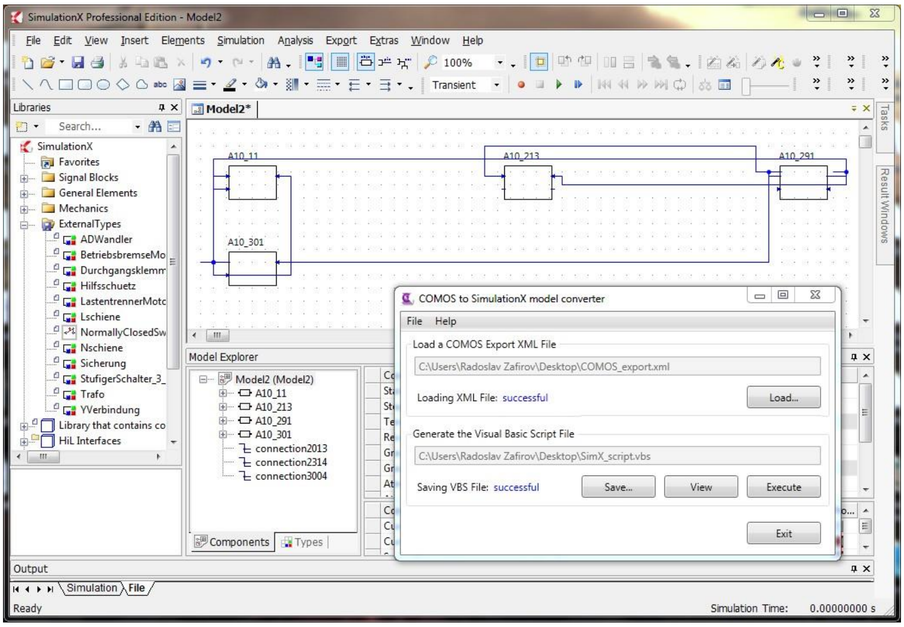

Figure 3. A model generated in SimulationX from an E-CAD model in COMOS with the demonstrator application “COMOS to SimulationX model converter”

A sample model, generated with the application is shown on Figure 3. The three different levels, on which the model is built, can be easily found in this example. The “External Types” library (Figure 3 to the left) with components containing device behavior modeled in Modelica represents the **model component level**. These same components are instantiated in the plant model (Figure 3: “Model2”) (**plant model level**) and can be found within compounds (Figure 2, Figure 3: “A10_11”) that resemble pages of electrical circuit diagrams (**model hierarchization level**).

With this model the relay logic of the electric drives in the conveyor boom can be simulated.

Summary and outlook

Generating interdisciplinary simulation models for virtual commissioning involves extra effort. This paper proposes an approach for automatic generation of the parts of such models that relate to electrical design. The models are generated in the multiphysics simulation language Modelica which allows for coupling or extension with models that relate to mechanical design for visualization purposes. The automatic model generation involves the transfer of model topology (instantiated components and the relations between them) from semantically rich electrical design models to interdisciplinary simulation models.

The following areas for possible future work have been identified:

Depending on the scope of virtual commissioning, it is not always required to transfer the complete system topology to the simulation model. Transferring information only on vital sensors and actuators can also lead to a simulation of the plant’s behavior. Categorizing electrical devices as vital and non-vital for simulation and implementing filters for model generation can help create manageable simulation models for complex automated plants, omitting hundreds of passive, non-critical components, like e.g. wire clamps.

The demonstrator application shown in this paper was created as proof of concept and not for active daily use. Modern techniques from Software Engineering such as the Model-Driven Engineering’s Model Transformation can be deployed for the implementation of the method shown here.

References

[1] Boucher, M.: System Design: New Product Development for Mechatronics, Benchmark Report, Boston, MA: Aberdeen Group, 2008

[2] Verein Deutscher Ingenieure (Hrsg.): VDI-Richtlinie, Simulation von Logistik-, Materialfluss und Produktionssystemen – Maschinennahe Simulation, Beuth Verlag, Düsseldorf, 2007

[3] Pritschow, G.; Röck, S.: “Hardware in the Loop” Simulation of MachineTools, CIRP Annals - Manufacturing Technology, Volume 53, Issue 1, 2004, Pages 295–298

[4] Böhme, T.; Kennel, M.; Schumann, M.; Winge, A.: Automatisierte Erstellung domänenübergreifender Modelle und echtzeitfähige Kopplung von Simulation, Visualisierung und realen Steuerungen, Paderborner Workshop Augmented & Virtual Reality in der Produktentstehung,8.2009, Paderborn, ISBN: 978-3-939350-71-2, p.155-172

[5] Dominka, S.; Heuschmann, C.: Prozessgutsimulation und Anlagenteilsimulation zur Verkürzung der Inbetriebnahme, atp - Automatisierungstechnische Praxis 49 (2007) 3, p. 55-60

[6] Kühne, T.: What is a Model?, Dagstuhl Seminar 04101, Dagstuhl, March 2005

[7] Stachowiak, H.: Allgemeine Modelltheorie, Springer-Verlag, Wien and New York, 1973

[8] Wünsch, G.: Methoden für die virtuelle Inbetriebnahme automatisierter Produktionssysteme (Dissertation): Institut für Werkzeugmaschinen und Betriebswissenschaften (iwb), Technische Universität München. München (07.12.2007)

[9] Reinhart, G.; Lacour, F.-F.: Physikbasierte mechatronische Simulation materialflussintensiver Produktionsanlagen, 7. Paderborner Workshop "Entwurf mechatronischer Systeme", ISBN: 978-3-93935-091-0, Paderborn, 2010, p. 85-97

[10] Siemens AG, Industry Sector, Industry Solutions Division: Simit: Die offene Simulationsplattform mit dem enormen Leistungsspektrum, www.siemens.de/simit, Siemens AG, 2009, last visit: 16.04.2012

[11] Heinemann, R.: PSPICE: Einführung in die Elektroniksimulation, Carl Hanser Verlag GmbH & CO. KG, Auflage 7, (05. Mai 2011), ISBN-10: 3446426094, ISBN-13: 978-3446426092

[12] Tiller, M. M.: Introduction to Physical Modeling with Modelica, Springer US, (31. Mai 2001) ISBN-10: 0792373677, ISBN-13: 978-0792373674

[13] Fritzson, P.; Engelson, V.: Modelica – A unified object-oriented language for system modeling and simulation, ECOOP’98 – Object-Oriented Programming; Lecture Notes in Computer Science, 1998, Volume 1445/1998, 67-90, DOI: 10.1007/BFb0054087

[14] Zickert, G.: Elektrokonstruktion - Engineering mit EPLAN, Carl Hanser Verlag GmbH & CO. KG, Auflage 2, neu bearbeitete Auflage, (5. Februar 2009), ISBN-10: 3446414703, ISBN-13: 978-3446414709, p.141

[15] Siemens AG: Comos Automation: Effizientes Engineering von der EMSR-Planung bis zur Automatisierungstechnik, 11.2011, www.siemens.com/comos, last visit: 16.04.2012

[16] International Electrotechnical Commission: IEC 61131-3 Programmable Controllers - Part 3: Programming Languages; Revision / Edition: 2.0; 2003

[17] Technical Operations, International Council on Systems Engineering (INCOSE): Systems Engineering Vision 2020; Document No.: INCOSE-TP-2004-004-02 Version/Revision: 2.03 Date: September 2007

[18] Verein Deutscher Ingenieure: VDI/VDE 3682: Formalized Process Descriptions, Beuth Verlag, Düsseldorf 2005

[19] Lorenz Hundt: International Standardisation Takes Next Step, AutomationML e. V. c/o IAF, www.automationml.org, 17.11.2011 (last visit: 17.04.2012)

[20] Dipl.-Ing. Mathias Maurmaier, M.Sc., Dipl.-Ing. Michael Lebschi, Prof. Dr.-Ing. Dr. h. c. Peter Göhner: Modelltransformation zur konsistenten Generierung von Schaltplänen und Quellcode aus Matlab/Simulink-Modellen. 7. Paderborner Workshop Entwurf mechatronischer Systeme, Paderborn, 2010