Object-Oriented Modelling in the Context of Networked Simulations

Ulrich Kramer, Christian Schröder
Competence Platform Networked Simulations
University of Applied Sciences Bielefeld

Introduction

In 2003, a novel institution was founded at Universities of Applied Sciences in the state of North Rhine-Westphalia which is explicitly devoted to applied research: the so-called competence platform. The background is that, due to the Bologna process which was then being implemented in North Rhine-Westphalia, an institutionalisation of research efforts was required in view of Bachelor and Master Courses, in order to bridge the gap to the established universities where research and education ideally stand in harmonic unity since Humboldt's days.

The topic of one of those institutions that our University has successfully applied for is the Competence Platform Networked Simulations. Ten colleagues from three engineering science departments mainly engaged in computer simulation allied to one research group with the objective to coordinate their research activities and to identify innovative application areas lying in between the conventional borders of various engineering disciplines.

What is the meaning of this made-up term Networked Simulations? The original vision was taken over from the idea of the digital factory where products and production are seamlessly to be designed, analysed, simulated, and tested in three-dimensional virtual reality by means of simulation tool chains long before they are put into real reality ready to be grasped with hands.

Research Demands

However, the harsh reality, in particular in small and middle size companies belonging to our preferred clientele, shows quite a different situation: the simulation tools are, unless ignored at all, often just isolated solutions. Each tool covers only a small fraction of the entire spectrum of the product and production simulation, respectively. In order to get closer to that vision of a digital factory, the tool users have to understand the principles of creating simulation tool networks by themselves. This is done typically in a way of point-to-point networking where the data exchange often has to be carried out manually (Fig. 1).

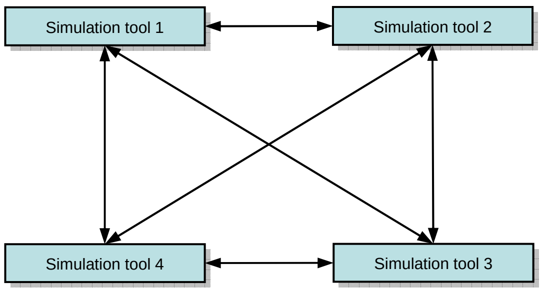

***Fig. 1** Conventional point-to-point networking of simulational tools*

Extending that approach to more and more simulational tools unfortunately leads to unmanageable complexity (Fig, 2).

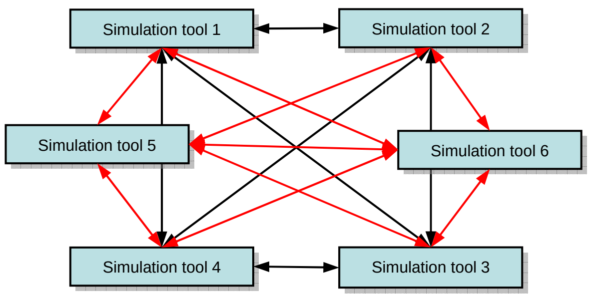

***Fig. 2** Unmanageable complexity of point-to-point networking of simulational tools*

In fact, it results in a distributed, redundant, and ultimately chaotic data management. Due to the "fixed wiring" of point-to-point networking, the simulation processes become maximally inflexible. Potential synergy effects remain unused, the maintenance costs are exploding. The consequences are frustration and acceptance problems in the companies.

Thus, we are far away from Computer Integrated Manufacturing or digital factories.

Competence Platform Mission

The conclusion drawn from market analyses, discussions with companies, and our own experiences has set the mission and the main activities of our Competence Platform: the focus on the design and development of instruments and methods for utilisation and optimization of potentials unused so far which result from intelligent networking of simulations over the entire product life cycle – reaching from very early ideas until recycling (Fig. 3).

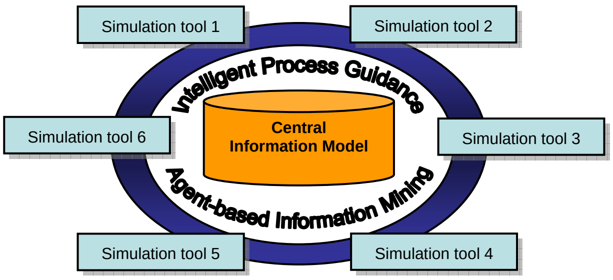

***Fig. 3** Basic networked simulations concept*

Four innovations fields have been identified: besides classical application areas of simulation, such as

- product simulation, with 3D-CAD modelling, finite elements methods, multi-body dynamics, and the like,
- process simulation, with material flow and manufacturing simulation tools, and
- test simulation, including reverse engineering, hardware-in-the-loop, or yield management simulation.

Common tasks to be accomplished have been defined, for example product, process, and engineering data management, or project and quality management. Thus, the horizon has been broadened from pure engineering to business processes. Principally, this change of perspective meets a distinct tendency towards the integration of enterprise computer applications as a whole.

Key Concepts of Networked Simulations

The information model forming a sustainable basis for most of our research and development activities is shown in Fig. 4.

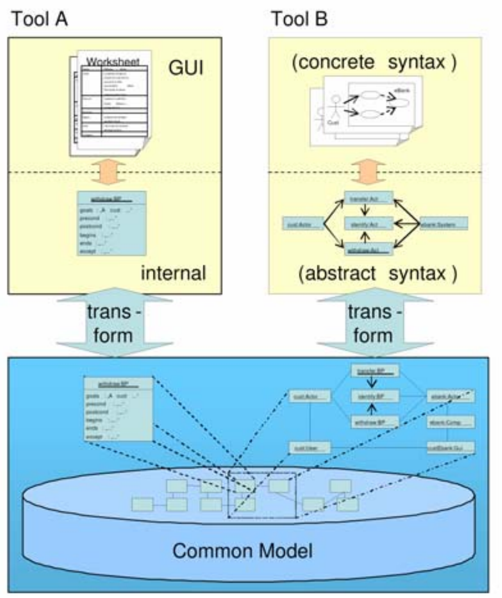

***Fig. 4** Information model of networked simulations*

Generally, this information model takes into consideration international data format standards such as STEP, though it leads to data representations which are open, central, scalable, and simulation tool independent. Starting with tool specific representations of the respective sub-models a model-based transformation is performed in order to achieve tool independent representations that can be pasted into the central information model.

In this regard, the XML/XMI coding techniques play an important role. XML stands for eXtensible Markup Language which was designed to be self-descriptive and to carry data by means of structuring, storing, and transporting rather than just displaying, XMI (XML Metadata Interchange) on the other hand is a way to save any MOF-based (MOF: meta object facility) models such as UML (Unified Modelling Language) representations in XML.

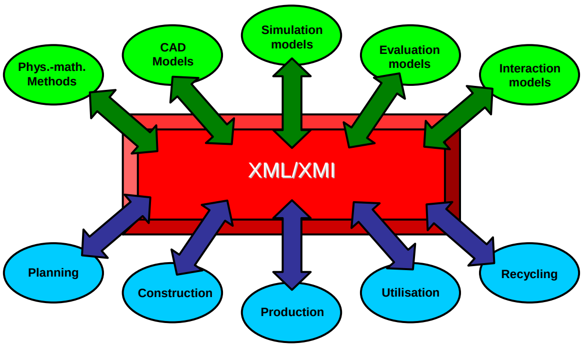

Fig. 5 *XML/XMI for networked simulations*

Using XML/XMI the abstraction of models generated by various sources such as CAD, simulation, or evaluation platforms may be accomplished. This leads to an increasingly intelligent simulation process guidance with computer aided process optimisation (i.e. computer aided design of optimisation strategies with respect to simulation parameters, e.g. material properties, costs, etc.). It is also planned to develop process templates for fast and individual product realisations in enterprises.

Sample Projects of the Platform

In some of our projects we were concerned with the detection of weak spots in enterprise procedures and analysis of the respective software tools. Often object parameters get lost as a result of model data export, e.g. dimension data is removed from CAD models when translated to FEM models, or reversely, mesh grid data is missing when CAD models are re-imported into the FEM system.

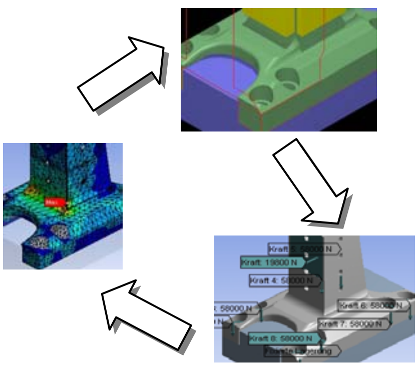

Fig. 6 *CAD – FEM product optimisation circle*

Some tools are behaving in that strange manner even if one and the same model is exported and re-imported in any available external format. Obviously, such effects impair the benefit of optimisation circles (Fig. 6) and, consequently, the acceptance of simulation tools. In such cases we analyse structures and formats of model data and try to simplify the transfer process.

Meanwhile, we are dealing not only with harmonisation of data flows from one engineering platform to another, but also with networking design platforms in connection with ERP (Enterprise Resource Planning) and PMC (Production Management and Control) systems (Fig. 7).

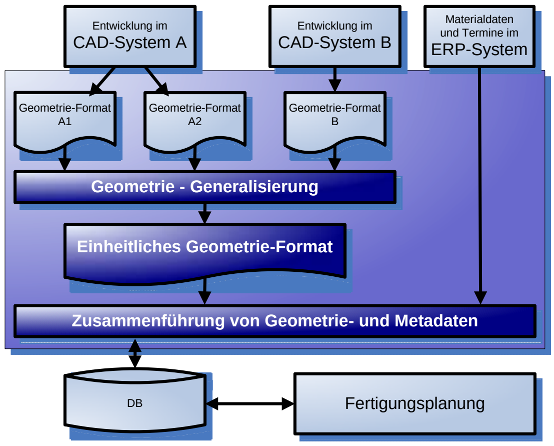

Fig. 7 *Product and process data management*

The tasks we had to cope with in one of our projects were 1) extraction and generalisation of the 3D geometric product data produced by diverse, mutually incompatible CAD systems, 2) generation of information models in a uniform geometry format, 3) fusion of engineering data with orders management and stock-keeping data by an ERP system, 4) generation of command parameters of the PMC system. In that case, the execution of according services was realised by means of public-domain, open source middleware modules, and the target containing the common information models was an Oracle database application. By extending our research efforts to the level organizational processes we are at present also dealing with the analysis, integration and optimisation of business processes – without any engineering references.

Object-Oriented Simulation Models

Object-oriented software structures are implicitly mentioned when we are dealing with XMI in networked simulations as a way to handle MOF-based model data. However, the relevance of object-orientation in the context of simulation models is not obvious but has a great impact on modelling languages and processes as we will show in the following paragraphs.

When we are talking about classes and objects we associate with this concept abstract data types which are built-up in an interface-oriented way characterized by names, attributes, and operations. These objects may be composed according to information hiding principles so that we can take the advantage to distinguish between public and private attributes or operations. Furthermore, we can simplify the implementation considerably when we utilise inheritance mechanisms.

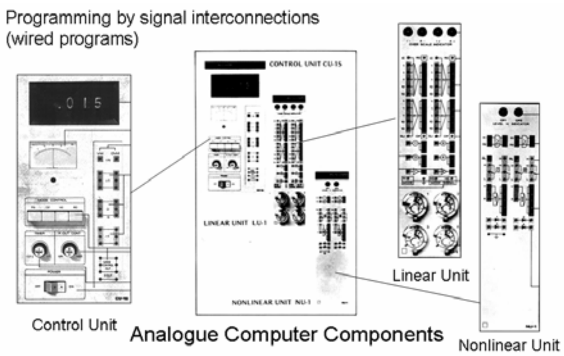

Fig. 8 *Analogue computer modules*

The software of early digital simulation tools simply adopted modules which were used to be applied already in analogue computers (Fig. 8). The programs of analogue computers were just the cables which interconnected amplifiers, integrators, potentiometers, etc. Accordingly, also digital simulation programs were organised like wired programs of analogue computers by signal interconnections of those modules so that one might actually be wondering what the virtues of object-oriented modelling should be in the area of dynamic simulation models. On the basis of such modules realised by a few subprograms it was possible to implement any systems of ordinary differential equations. The consequence is a predominance of signal flow models as shown in Fig. 9 in the representation of dynamic models.

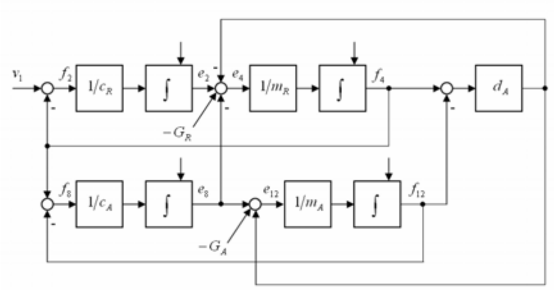

Fig. 9 *Example of a signal flow model (linear quarter vehicle vertical dynamics)*

Towards Component-Based Models

The advantages of signal flow models are evident: 1) direct transformation of mathematical descriptions into simulation models, 2) output independency of input signals, 3) local impacts of modelling errors, and 4) simple analysis. Signal flow modelling, however, also has remarkable disadvantages: before we can start a model transformation, we need its mathematical representation; furthermore, energy feedback effects occurring when physical components are interconnected have to be explicitly modelled.

Fig. 10 shows a simple example of two resistor-capacitor elements that are to be linked. In the iconic model representing the components and their interconnection is simply carried out by linking together the corresponding wires. Since any model view is just an abstraction from the real physical situation, in the signal flow model the load of the second RC-element has to be explicitly modelled by adding a feedback block. Thus, it is impossible to gain a simulation model from the iconic representation by applying simple transformation rules. This is the price one has to pay for the advantage of decoupling of input and output signals.

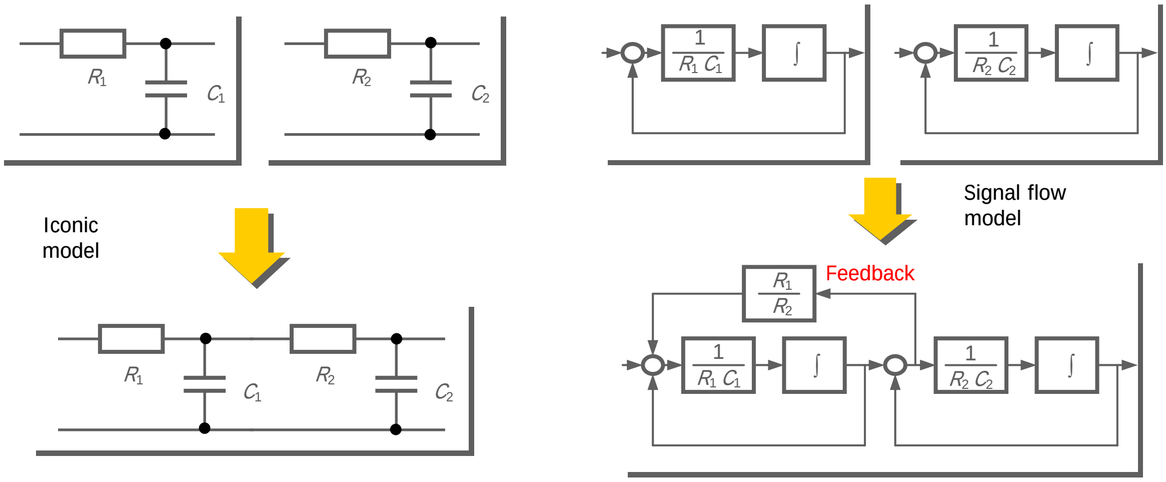

Fig. 10 *Connecting two RC-elements in iconic and signal flow model representations*

Energy Flow Modelling

In the 1960s, several MIT researchers were active in attempts of improving the plausibility of dynamic simulation models and simplifying the process of modelling itself. Most notably, Jay Forrester developed his system dynamics concept on the basis of "states and rates", leading him and his coworkers to the elaboration of world dynamics models. Henry Paynter invented the bond graphs which were originally aimed at the representation of engineering systems on paper by letter elements and energy flows between them.

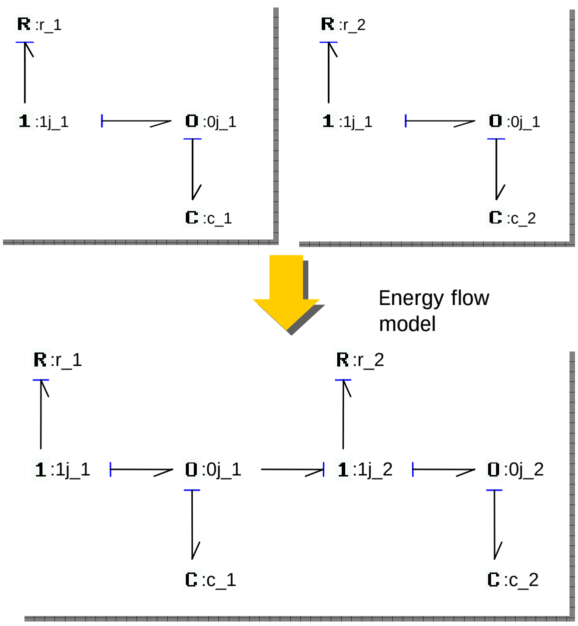

Fig. 11 *Connecting two RC-elements in energy flow representation*

Like signal flow models, bond graphs are abstractions from the relevant physical nature of engineering components, but they retain physics insofar as they generalise the energy exchange of the components. This means, that all one-port elements (elements with two links) are classified to be either sources (energy import), consumers (energy export), or (energy) storages. Additionally, elements with two ports are defined as transformers or gyrators which are power neutral, i.e. the power at both ports is equal. Finally, elements with two and more ports are defined as so-called junctions which join and distribute the energy flows.

Due to the identity of flow and rate, energy flows are in fact power distributions. Accordingly, each of the energy flow symbols, the bonds (half head arrows, Fig. 11), represents two physical variables. In terms of bond graphs these variables are called effort and flow. Depending on their particular physical meaning we can associate them with different energy forms, e.g. electrical, magnetical, mechanical translational, mechanical rotational, thermal, or fluidic. The product of these pair-wise occurring variables (conjugate variables) has the physical dimension of power. The power conservation principle allows mixing of different energy forms thus bond graph modelling is best suited for multi-domain engineering systems.

Bond Graph Modelling Examples

Electric and magnetic energy are coupled by Maxwell's equations in integral form:

$$\oint H\,\mathrm{d}l = \Theta \qquad \oint E\,\mathrm{d}l = -\frac{\mathrm{d}\Phi}{\mathrm{d}t}$$

The magnetomotive force $\Theta = N\,i$ is the magnetic effort variable, and the voltage u is the time derivative of the flux linkage $\lambda = N\Phi$, so that it is related with the induction rate $\dot{\Phi}$ by $u = N\dot{\Phi}$ (N: number of conductor turns). Thus, the transition from electric energy to magnetic energy is described by a gyrator (Fig. 12).

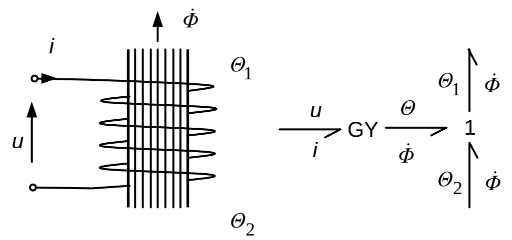

***Fig. 12** Domain transitions between electric and magnetic energy form*

The technical realisation of an electromagnetic converter, for example, is modelled by two gyrators each describing the electric-magnetic domain transition of both the primary and the secondary circuit, and two C-storages (corresponding to capacitors in the electric energy domain) are needed to express Hopkinson's law (Fig. 13).

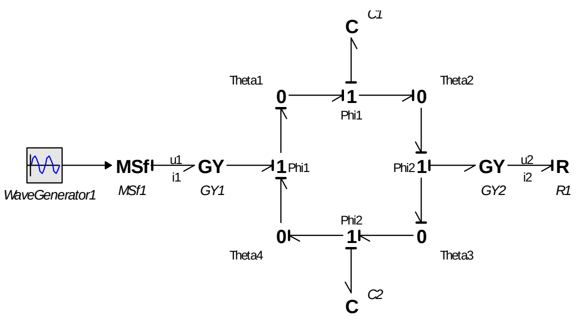

***Fig. 13** Bond graph model of a technical converter*

It is worth to be noted, that Hopkinson's law postulates a linear relation of magnetomotive force Θ_k and the magnetic induction Φ_k in the k-th section of a magnetic circuit:

$$\Theta_k = R_k \cdot \Phi_k\,,$$

where R_k is used to be interpreted as the magnetic resistance. However, from a bond grapher's point of view the alleged magnetic resistance (or more precisely the magnetic conductance) emerges as a C-storage given by the equation

$$\dot{\Phi} = \frac{1}{R} \cdot \frac{\mathrm{d}\Theta}{\mathrm{d}t}\,.$$

So sometimes bond graph modelling may offer unexpected insights into the inner structure of physical relations.

Another example is the electromagnetic levitation setup depicted in Fig. 14. The energy in the coil with inductivity, L, and current i is given by

$$E = E(x,i) = \frac{1}{2} L(x)\, i^2$$

as a function of both variables, i and x.

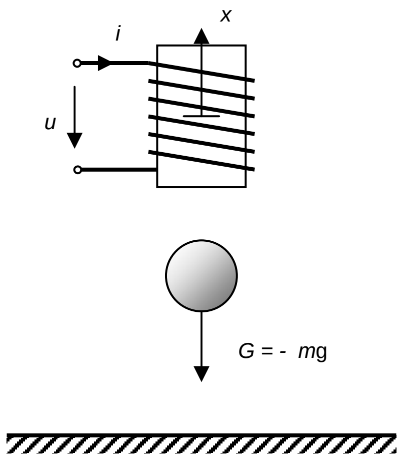

***Fig. 14** Experimental setup of a levitation experiment*

In this case, the system cannot be decomposed into magnetic and mechanical domains, since the total derivative of the energy E

$$\mathrm{d}E = \frac{\partial}{\partial i} E(x,i)\,\mathrm{d}i + \frac{\partial}{\partial x} E(x,i)\,\mathrm{d}x$$

is not a sum of total derivatives of single energy forms. The partial derivatives of E can be identified with the following physical variables:

$$\varphi = \frac{\partial E}{\partial i} = L(x)\,i \qquad \text{magnetic induction}$$

$$F = \frac{\partial E}{\partial x} = \frac{i^2}{2}\frac{\partial L(x)}{\partial x} \qquad \text{mechanical force}$$

The above expressions are related by

$$\frac{\partial \varphi}{\partial x} = \frac{\partial F}{\partial i} = \frac{\partial^2 E}{\partial x \partial i}$$

which do not vanish. In order to achieve an integral causal model, further rearrangements of the equations are needed. The resulting system of nonlinear ordinary differential equations is given by:

$$\frac{\mathrm{d}i}{\mathrm{d}t} = \frac{1}{L}u - \frac{i}{L}\frac{\partial L}{\partial x}v$$

$$\frac{\mathrm{d}F}{\mathrm{d}t} = \frac{i}{L}\frac{\partial L}{\partial x}u + \left(\frac{i^2}{2}\frac{\partial^2 L}{\partial x^2} - \frac{i^2}{L}\left(\frac{\partial L}{\partial x}\right)^2\right)v$$

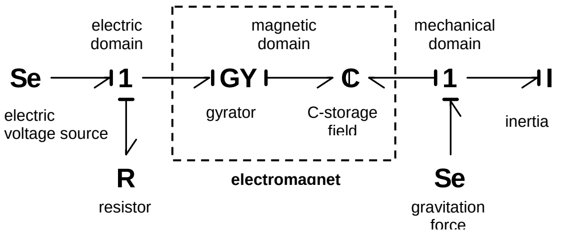

Fig. 15 *Bond graph representation of an electromagnet*

The bond graph representation is depicted in Fig. 15. It encloses three physical domains and it contains a C-storage field to cope with the impossibility to separate the magnetic and mechanical domain.

Potentials and Constraints of Symbolic Modelling

Starting with an iconic diagram the graphic modelling process takes different directions: When a signal flow model is aspired then at first the iconic model has to be analysed by means of balance principles such as Kirchhoff's rules in order to generate a mathematical model before the corresponding signal flow model can be implemented (Fig. 16).

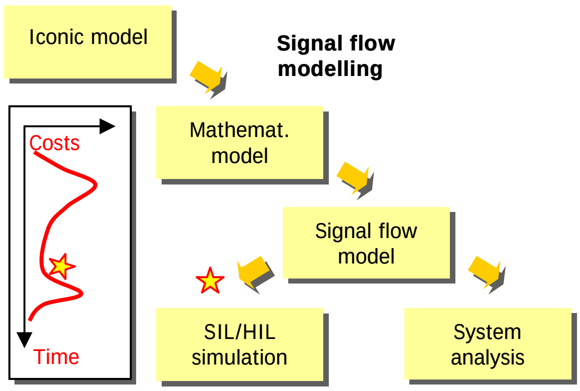

Fig. 16 *Signal flow modelling process*

For setting up energy flow models, however, we can in most cases translate iconic models directly into bond graphs. Provided that a suitable simulation tool is available, a mathematical model is not needed until the system has to be analysed in order to design controllers or observers, for example.

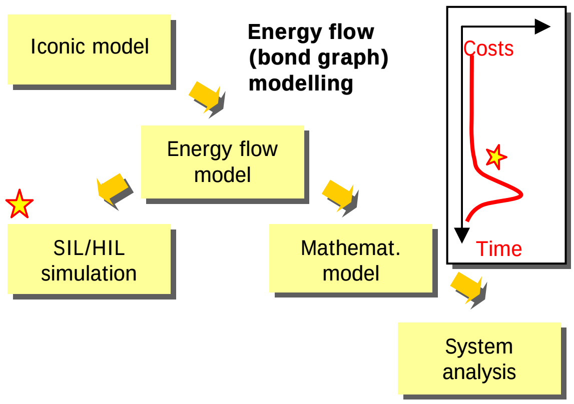

Fig. 17 *Energy flow modelling process*

Following a suggestion by Gawthrop (1994) we made first attempts to bypass mathematics even in designing symbolically model based observers (Fig. 18) and controllers (Fig. 19).

It should be noted that despite the emphasized differences of signal flow models and energy flow models, the barrier between them is not unconquerable. Both representations allow, after all, the simulation of respective systems, thus it ought to be possible to specify rules by which energy flow representations can be translated into equivalent signal flow models. In fact, it is not difficult to formulate such rules and this is at present exactly the way we familiarise students with techniques of dynamic systems modelling.

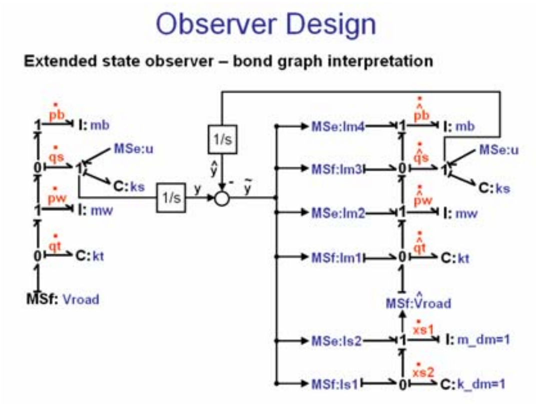

Fig. 18 Bond graph based observer design (Pittner, 2006)

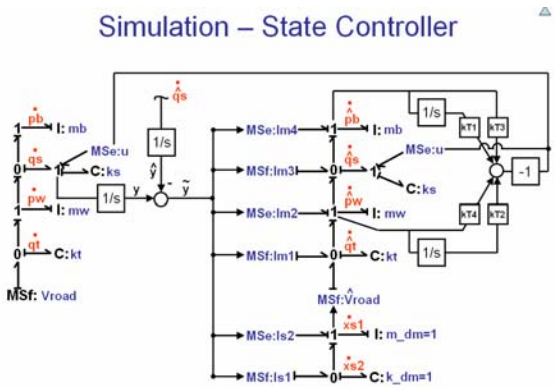

Fig. 19 Bond graph based controller design (Pittner, 2006)

Conclusions for Object-Oriented Component Based Modelling

Component based models of engineering systems are used as preferred representations in most technical disciplines where they have been developed in a domain specific manner mostly derived from familiar iconic representations. Meanwhile, mechatronic systems design requirements (e.g. VDI 2206 Guideline) suggest model-based design procedures and force multidisciplinary system representations with component models (Fig. 20).

An important precondition for component based modelling is, of course, the identification of elements such as bodies, joints, and hinges in the case of mechanics, for example, and to implement them in domain specific class libraries. Additionally, a few rules are needed to achieve integral causality which is essential for using models in a real-time environment. In this respect, most of commercial modelling tools including Modelica and Dymola, respectively, have in common the efforts to develop and enlarge such class libraries.

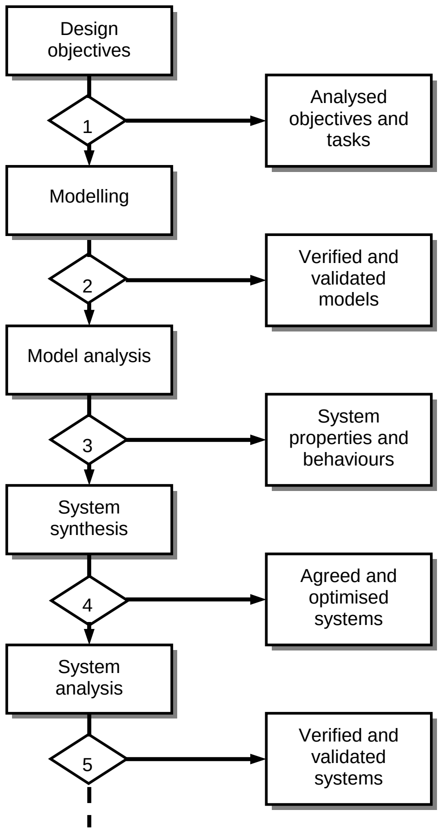

Fig. 20 Model-based design of mechatronic systems (after VDI 2206 Guideline)

Unfortunately, energy flow modelling does not automatically ensure component based model representations. It has been demonstrated that this requirement is a subject to the separability of energy forms and inasmuch depending on the underlying physics. This is the reason why object-oriented, component based modelling is a never-ending story and will remain a persistent challenge on the way towards model-based engineering.

Summary

The preferences for component based modelling of engineering systems in the context of networked simulations over other techniques can be summarized in a series of advantages: 1) component based modelling utilizes accustomed domain specific iconics, 2) it allows incremental modelling, 3) component based models can be easily modified and refined, 4) the interfacing of domain elements is simple.

At a first glance, iconic representations seem to be the convenient choice for communication, though they are used to be results of demanding abstractions. Even in intuitively plausible appearing mechanical networks, for example, one is searching in vain for the second terminal of the one-port element inertia. One has just to know that this terminal is – invisibly – connected to the resting reference system (inertial system).

Modelling is for sure an inevitable preparational task before doing any simulation, but utilizing the simulational results afterwards is yet another. Due to the rapid changes in the simulational software market it is almost impossible to run a simulation after a long time under exactly the same conditions as they existed in the beginning. What we need to overcome this problem is a sort of simulation data reinforcement which allows us to archive and to retrieve simulation parameters and results independently from actual software simulation tools. However, such simulation data reconstruction may require novel linguistic concepts for the development of agent-based information mining.

We conclude that still a lot of time and effort is needed to define simulation models for multidisciplinary and interdisciplinary communication as well. Hardware based data storage is sometimes interpreted as externalised human thoughts. Accordingly, we are free to interpret simulation models as externalised dynamic processes. However, before we can benefit from such externalised dynamic processes, especially in the context of Networked Simulations, we have to cope with a lot of unsolved problems. Let's start!

References

Gawthrop, P.J. Physical model-based control: a bond graph approach. Journal of The Franklin Institute, Volume 332, Number 3, May 1995, pp. 285-305(21)

Pittner, A. Bond graph based design of an extended state observer. Master Project, University of Applied Sciences Bielefeld, 2006.

Acknowledgement: The Competence Platform Networked Simulations is funded by the Ministry of Research, State of North Rhine-Westphalia, Düsseldorf (decree of 10.12.2004).

Authors: Ulrich Kramer, Speaker of the Competence Platform Networked Simulations, Professor of Automation and Control Engineering, University of Applied Sciences Bielefeld, 1974 Ing. (grad.) (Gyrators in active filters) Technische Fachhochschule Berlin (formerly Gauss Akademie), 1978 Dipl.-Ing. (Application of the maximum likelihood method to signal analysis) Technische Universität Berlin, 1985 Dr.-Ing. (Mobile systems analysis in simulations and experiments) Technische Universität Berlin, 1985-1990 Product Research Manager (Vehicle Cybernetics) BMW, Munich, 2000 Visiting Scientist (Performance analysis of driving simulators) DaimlerChrysler Berlin, since 2002 Visiting Lecturer (Vehicle and driving dynamics) Technische Universität Cottbus.

Christian Schröder, Deputy Speaker of the Competence Platform Networked Simulations, Professor of Applied Mathematics and Software Engineering, University of Applied Sciences Bielefeld, 1995 Dipl.-Phys. (Current transport in SiC Schottky contacts) Universität Osnabrück and Daimler-Benz Forschung Frankfurt am Main, 1999 Dr. rer. nat. (Numerical simulations of thermodynamic properties of magnetic materials) Universität Osnabrück and Ames Laboratory, Ames, Iowa USA, 1999 Scientist (Medical image reconstruction for CT and MRI) Philips Research Laboratories, Hamburg, 2000-2002 Scientist (Software and Systems Modelling) Telelogic AB, Bielefeld, 2002-2003 Scientist (Simulation of Magnetic Molecules) Ames Laboratory, Ames, Iowa, USA.

Contact: Prof. Dr.-Ing. Ulrich Kramer
Phone: +49(0)521 106-7208
Email: ulrich.kramer@fh-bielefeld.de