EDITORIAL

Models in simulation

Jeff Gray[1] · Bernhard Rumpe[2]

In the presence of increasing complexity, simulation has become a vital tool for obtaining additional information about a subset of the world, such as the interactions of all entities, processes, life forms, and other key components that aid in understanding a particular context. A key advantage of simulation is the Principle of Substitution, which is perhaps best summarized by a quote from Marvin Minsky:

> We use the term "model" in the following sense: To an observer B, an object A^* is a model of an object A to the extent that B can use A^* to answer questions that interest him about A. It is understood that B's use of a model entails the use of encodings for input and output, both for A and A^*. If A is the world, questions for A are experiments. A^* is a good model of A, in B's view, to the extent that A^*'s answers agree with those of A's, on the whole, with respect to the questions important to B. (Minsky 1965)

The following definition offers several insights into the benefits of simulation and the importance of a model toward understanding parts of a real system:

> Simulation is the imitation of the operation of a real-world process or system over time. The act of simulating something first requires that a model be developed; this model represents the key characteristics or behaviors/functions of the selected physical or abstract system or process. The model represents the system itself, whereas the simulation represents the operation of the system over time. Simulation is used in many contexts, such as simulation of technology for performance optimization, safety engineering, testing, training, education, and video games. Often, computer experiments are used to study simulation models. Simulation is also used with scientific modeling of natural systems or human systems to gain insight into their functioning. (Wikipedia 2016)

The definition further states the benefits and usefulness of simulation:

> Simulation can be used to show the eventual real effects of alternative conditions and courses of action. Simulation is also used when the real system cannot be engaged, because it may not be accessible, or it may be dangerous or unacceptable to engage, or it is being designed but not yet built, or it may simply not exist. (Wikipedia 2016)

The core topics and ongoing research questions driven by simulation include the following:

> Key issues in simulation include acquisition of valid source information about the relevant selection of key characteristics and behaviors, the use of simplifying approximations and assumptions within the simulation, and fidelity and validity of the simulation outcomes. Procedures and protocols for model verification and validation are an ongoing field of academic study, refinement, research and development in simulations technology or practice, particularly in the field of computer simulation. (Wikipedia 2016)

Modeling is a key ingredient for all forms and purposes of simulation. While not all forms of simulation are concerned

✉ Bernhard Rumpe
 bernhard.rumpe@sosym.org

 Jeff Gray
 jeff.gray@sosym.org

[1] University of Alabama, Tuscaloosa, Alabama, USA

[2] RWTH Aachen University, Aachen, Germany

with computer systems, most applications have some computational context. When considering the high-performance computing (HPC) community as an example, it can be observed that HPC researchers practice forms of modeling quite often. However, they very rarely use explicit modeling languages, but rather encode their models in low-level, but very efficient programming languages, such as C and Fortran, or they may even adopt Python as an easy to use configuration language for their high-level configuration.

Simulation science is emerging as a new discipline that connects a large set of sub-disciplines from many areas of domain knowledge. However, a simulation technology should have a mathematical and computer science foundation to handle the forms of models usable for simulation. The math theory on derivation/integration and the digital theory defined by computer science, which is based on discrete state changes in various forms, are core requirements. Numerical and algebraic techniques, software engineering, software architecture, compiler construction, computer architectures, networking, data science, machine learning, and numerous other related disciplines are necessary to fully use high performance computers and give users of various domains the desired tools that allow the application of simulation to new problem contexts.

Although modeling is a core element of simulation, it is not clear how explicit models will need to be in the future of simulation. Currently, the driving force is still the efficiency of the running program, while the effectiveness of the developer plays a lesser role. This was true for the automotive domain 10 years ago, which relied on C and assembly code for core computation expression. However, the automotive domain evolved—partially because better and more efficient compilers are available—and adopted more higher-level modeling languages for designing and implementing automotive software. It is not yet clear whether this same trend will occur in other areas, such as HPC. Early examples show that this could work. For example, one of the authors of this editorial has defined a modeling language for neurons and brain activities, now in use in the European brain project.

When simulation becomes more dominant and broadly used by various industries, then there will likely be a need for additional focus on the effectiveness of developers, and many other-ilities, such as reusability of the developed simulations. More advanced simulation techniques will also drive the need for explicit modeling languages and appropriate tooling for generating and deploying efficiently running code, for gathering and analyzing the simulation data, for understanding the deviations between simulation and reality and for predicting the future.

Edward Lee's keynote at the MiSE (Modeling in Software Engineering) workshop at ICSE 2016 underscores the importance of models and their various uses in both science and engineering: "If the model does not describe the reality, then to a scientist, the model is wrong, but to an engineer the constructed reality is wrong." If we want to understand whether a model and its reality correspond, simulation is one of several techniques that can aid in understanding complexity.

We hope to see more papers on modeling for simulation and simulation of models in future SoSyM submissions!

References Cited

- Marvin Minsky, "Matter, Minds, and Models," *International Federation of Information Processing Congress*, 1965, vol. 1, pp. 45–49.
  (also available at http://groups.csail.mit.edu/medg/people/doyle/gallery/minsky/mmm.html)
- Wikipedia, "Simulation," http://en.wikipedia.org/wiki/Simulation, Last Accessed June 15, 2016.

Content of this issue

This issue contains an Expert Voice article and eleven regular papers, as enumerated below. The Expert Voice contribution emerges from the prestigious *Most Influential Paper Award* that was given to Dániel Varró and András Pataricza at MODELS 2014 for their paper "Generic and Meta-Transformations for Model Transformation Engineering," which was presented originally at UML 2004. The SoSyM version of this paper reflects on the history and the current state of model transformation research.

The paper by Eichler et al., was selected as one of the "SoSyM Journal-First" papers for 2015 (the other three SoSyM-First papers were published in the last issue). This paper was first submitted and accepted in the normal SoSyM paper selection process and then presented in October 2015 at MODELS 2015.

- **Expert Voice**
  "Road to a reactive and incremental model transformation platform: three generations of the VIATRA framework" by Daniel Varro, Gabor Bergmann, Ábel Hegedüs, Ákos Horváth, István Ráth, and Zoltán Ujhelyi
- "Comprehensive Analysis of FBD Test Coverage Criteria Using Mutants" by Donghwan Shin, Eunkyoung Jee, and Doo-Hwan Bae
- "Model Transformation Intents and Their Properties" by Levi Lucio, Moussa Amrani, Jürgen Dingel, Leen Lambers, Rick Salay, Gehan Selim, Eugene Syriani, and Manuel Wimmer
- **A SoSyM Journal-First paper**
  "Enhanced Graph Rewriting Systems for Complex Software Domains" by Cédric Eichler, Thierry Monteil, Patricia Stolf, Alfredo Grieco, and Khalil Drira
- "Turning Event Logs into Process Movies: Animating What Has Really Happened" by Massimiliano de Leoni, Suriadi Suriadi, Arthur H. M. ter Hofstede, and Wil M. P. van der Aalst

- "Query-driven soft traceability links for models" by Ábel Hegedüs, Akos Horvath, Istvan Rath, Rodrigo Starr, and Dániel Varró
- "Empirically Evaluating OCL and Java for Specifying Constraints on UML Models" by Tao Yue and Shaukat Ali
- "Least-change Bidirectional Model Transformation with QVT-R and ATL" by Nuno Macedo and Alcino Cunha
- "Clafer: Unifying Class and Feature Modeling" by Kacper Bak, Zinovy Diskin, Michal Antkiewicz, Krzysztof Czarnecki, and Andrzej Wasowski

- "Systematic Approach for Constructing an Understandable State Machine from a Contract-based Specification: Controlled Experiments" by Heung Seok Chae and Jung Ho Bae
- "Expressing Aspectual Interactions in Design: Evaluating three AOM approaches in the Slot Machine Domain" by Johan Fabry, Arturo Zambrano, and Silvia Gordillo
- "Feature-Based Classification of Bidirectional Transformation Approaches" by Soichiro Hidaka, Massimo Tisi, Jordi Cabot, and Zhenjiang Hu