VARIABILITY MODELING AND SIMULATION USING MULTIPLE SIMULATORS (TUTORIAL)

Thorsten Pawletta
Hendrik Folkerts
Christina Deatcu
Research Group of Computational Engineering & Automation
Wismar University of Applied Sciences
D-23966 Wismar, Philipp-Müller-Str. 14, Germany
{thorsten.pawletta, christina.deatcu}@hs-wismar.de, hendrik.folkerts@cea-wismar.de

ABSTRACT

The tutorial introduces variability modeling and simulation using multiple simulators based on an extended System Entity Structure and Model Base (SES/MB) architecture. It is shown how simulation models can be automatically selected, generated, and executed on the basis of a given design objective and how results of simulation runs can be evaluated. All concepts and methods are discussed using a case study. Along the way, the theory is practically demonstrated using a step-by-step implementation of the case study. Free software tools are introduced and their integration with OpenModelica is shown. Furthermore, the integration with MATLAB/Simulink is pointed out.

Keywords: variability modeling, model generation for multiple simulators, system entity structure, functional mock-up interface, OpenModelica, MATLAB/Simulink.

1 INTRODUCTION

In software engineering, variability is defined as the ability of a system to be configurable, extendable or adaptable depending on its purpose and objective. In order to master the problem of variability, the approach of software product line engineering (SPLE) was developed. SPLE refers to methods and tools for creating variants of similar software systems from a shared set of basic software.

Analogous to SPLE, variability management is a challenge in the area of modeling and simulation. A family of model variants has to be organized, from which a suitable candidate model has to be selected, generated, and evaluated depending on given objectives. The System Entity Structure and Model Base (SES/MB) framework introduced by Zeigler and colleagues provides a sound approach for such an organization. Based on the SES/MB approach, the authors developed an extended architecture (Folkerts, Pawletta and Deatcu 2020). Key points of the architecture are approaches: (i) to organize simulator-dependent/independent model bases, (ii) to generate models for multiple simulators, and (iii) to fully automate simulation experiments with model families based on given objectives. The focus of the tutorial is not on a complete discussion of all methods and extensions, but put on basic conceptual understanding of the SES/MB approach and the extended architecture.

2 STRUCTURE OF THE TUTORIAL

The tutorial provides a step-by-step introduction to variability modeling and simulation as well as to the automation of simulation experiments based on the SES/MB approach. Based on a continuous case study, the theoretical concepts are introduced and their software implementation is demonstrated. The tutorial is divided into six main topics and ends with a conclusion.

ANNSIM '21, July 19-22, 2021, Fairfax, VA, USA; ©2021 Society for Modeling & Simulation International (SCS)

2.1 The Case Study

Two structural variants of a control system are introduced. The aim is to automatically find the minimum possible control structure and its parameter settings for different control objectives via system simulation.

2.2 Basics of SES/MB based Modeling

The basics of modeling model families with the SES/MB approach is introduced. Subsequently, the simulator-independent specification of model variants, structures and parameter settings using an SES are discussed by means of the case study. Afterwards, the organization of simulator-dependent dynamic basic models in an MB is shown. For the case study, an MB is built with Modelica basic components.

2.3 Practical Modeling of an SES

Using the previously specified SES, practical modeling with a Python-based SES editor and its integration with a separate SES viewer is shown. Advanced SES methods are demonstrated in practice, such as the merging of SES models. For exchange with other tools, SES models are stored in JSON or XML file format. As an example of another tool, reference is made to a free SES Toolbox developed specifically for MATLAB/Simulink.

2.4 Model Selection and Model Generation

The derivation of concrete model configurations from an SES and the generation of simulator-specific models is shown. This is followed by a software demonstration for the case study, which results in the generation of an OpenModelica model.

2.5 Organization of Simulator-Independent Model Bases

Based on the Functional Mock-up Interface (FMI), an approach to build simulator-independent MBs is shown. This is demonstrated for the case study.

2.6 Full Automation of Simulation Experiments

An extended SES/MB architecture with components for a full automation of experiments is presented. Starting from an objective and an experiment specification, model configurations are derived from the SES, executable models are generated simulator-specific or FMI-based, executed with a simulator, and the simulation results are analyzed. Depending on the results, further model configurations are studied or the experiment is terminated.

2.7 Conclusion

Finally, existing deficits and possible further developments of the SES/MB approach as well as of our software tools are discussed in the context of current research. The demonstrated SES/MB-based software tools are available at github (https://github.com/cea-wismar/, https://github.com/hendrikfolkerts/).

ACKNOWLEDGMENTS

We thank Bernie Zeigler and Jerzy Rozenblit for the valuable discussions and the promotion of our research.

REFERENCES

Folkerts, H., T. Pawletta, and C. Ceatcu. 2020. “Model Generation for Multiple Simulators Using SES/MB and FMI”. *SNE-Simulation Notes Europe Journal* vol. 31(1), pp. 25-32. doi: 10.11128/sne.31.tn.10554