Chapter 4 Model-Based Development and Spatiotemporal Behavior of Cyber-Physical Systems

Peter Herrmann

Norwegian University of Science and Technology, Norway

Jan Olaf Blech Altran, Germany

Fenglin Han Norwegian University of Science and Technology, Norway

> Heinz Schmidt RMIT University, Australia

ABSTRACT

Many cyber-physical systems operate together with others and with humans in a joint physical space. Because of their operation in proximity to humans, they have to operate according to very high safety standards. This chapter presents a method for developing the control software of cyber-physical systems. The method is model-based and assists engineers with spatial and real-time property verification. In particular, the authors describe a toolchain consisting of the model-based development toolset Reactive Blocks, the spatial analyzer BeSpaceD in conjunction with the real-time model checkers UPPAAL and PRISM. The combination of these tools makes it possible to create models of the control software and, if necessary, simulators for the actual system behavior with Reactive Blocks. These models can then be checked for various correctness properties using the analysis tools. If all properties are fulfilled, Reactive Blocks transforms the models automatically into executable code.

DOI: 10.4018/978-1-5225-7268-8.ch004

Copyright © 2019, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.

INTRODUCTION

In safety critical domains like aviation, automotive and robotics, autonomous cyberphysical systems interact with each other and with humans in the same physical space. To avoid damage of machine equipment and injuries of humans, the control software of these systems has to guarantee spatiotemporal properties like collision avoidance or the reliable cooperation of several units that carry a heavy workpiece together. A popular way for the creation of functionally correct and safe system software is the application of integrated modeling and verification tools like MATLAB/Simulink (Tyagi, 2012). Our contribution is the combination of such a tool with efficient provers allowing engineers to verify that the coordinated behavior of multiple controlled cyber-physical systems fulfills relevant spatial safety properties. We introduce a toolchain combining the model-based engineering tool-set Reactive Blocks (Kraemer, Slåtten, & Herrmann, 2009) with the verification tool BeSpaceD (Blech & Schmidt, 2013). In particular, we use a development workflow starting with the collection of requirements for a cyber-physical system and its architecture followed by the steps 1 to 7 below:

- 1. Spatiotemporal properties of components are described in the input language of BeSpaceD.
- 2. A model of the system controller is created in Reactive Blocks. We compose it with a simulator model of the continuous system parts which is engineered using the BeSpaceD model developed in step 1.
- The built-in model checker of Reactive Blocks is used to check the combined controller and simulator model for general design errors, (Kraemer, Slåtten, & Herrmann, 2009).
- 4. If the checks in step 3 are passed, the software model is transformed into the input language of BeSpaceD.
- 5. Assuming certain maximum reaction times of the discrete controller, the resulting model is now verified with BeSpaceD to check whether it fulfills the required spatiotemporal properties defined in step 1.
- One of the model checkers UPPAAL (Bengtsson, et al., 1996) and PRISM (Kwiatkowska, Norman, & Parker, 2009) is now applied to prove that the realtime properties assumed in the proofs of step 5 are preserved by the Reactive Blocks model created in step 2 (Han & Herrmann, 2013), (Han, Herrmann, & Le, 2013).
- 7. By using the code generator from Reactive Blocks (Kraemer & Herrmann, 2007), (Kraemer, Herrmann, & Bræk, 2006), executable Java code of the controller and, if needed, of the simulator of the continuous behavior is created.

23 more pages are available in the full version of this document, which may be purchased using the "Add to Cart"

button on the publisher's webpage: www.igi-

global.com/chapter/model-based-development-and-

spatiotemporal-behavior-of-cyber-physical-systems/214832

Related Content

Integrated Design of eBanking Architecture

Tony C. Shanand Winnie W. Hua (2010). *Web Services Research for Emerging Applications: Discoveries and Trends (pp. 67-86).* www.irma-international.org/chapter/integrated-design-ebanking-architecture/41518

LocPSORank-Prediction of Ranking of Web Services Using Location-Based Clustering and PSO Algorithm

V. Mareeswariand E. Sathiyamoorthy (2018). *International Journal of Web Services Research (pp. 38-60).*

www.irma-international.org/article/locpsorank-prediction-of-ranking-of-web-services-using-location-based-clustering-and-pso-algorithm/205610

Measuring the Service Quality of E-Commerce and Competitive Strategies

Shu-Fong Chang, Jen-Chi Chang, Kuo-Hua Lin, Bin Yu, Yu-Cheng Lee, Sang-Bing (Jason) Tsai, Jie Zhou, Chao Wuand Zi-Chun Yan (2014). *International Journal of Web Services Research (pp. 96-115).*

www.irma-international.org/article/measuring-the-service-quality-of-e-commerce-andcompetitive-strategies/122817

Technical Architecture of Enabling Body of Knowledge System for Effective Learning and Information Dissemination

Liang-Jie Zhangand Jia Zhang (2013). *International Journal of Web Services Research (pp. 41-62).*

www.irma-international.org/article/technical-architecture-of-enabling-body-of-knowledge-systemfor-effective-learning-and-information-dissemination/90265

Geospatial Web Services for Distributed Processing: Applications and Scenarios

Theodor Foerster, Bastian Schäffer, Bastian Baranskiand Johannes Brauner (2011). *Geospatial Web Services: Advances in Information Interoperability (pp. 245-286).* www.irma-international.org/chapter/geospatial-web-services-distributed-processing/51490