Book Review

Building Software for Simulation: Theory and Algorithms, with Applications in C++

Author: James Nutaro

Bernard P. Zeigler (zeigler@rtsync.com)
Chief Scientist, RTSync Corp., 530 Bartow Drive Suite A Sierra Vista, AZ 85635
Professor Emeritus, Arizona Center for Integrative Modeling and Simulation
Research Professor, C4 I Center, George Mason University

“Building Software for Simulation: Theory and Algorithms, with Applications in C++ “1 is true to its name combining theory with theory implementations, all illustrated by applications. The theory is an updated and abridged rendering of that presented in “Theory of Modeling and Simulation” the second edition of which came out over a decade ago[1]. Readers who would like to follow this book’s development in depth may want to go back to the more leisurely and comprehensive development of the original. That said, the book advances existing theory in important ways. For example, it expands the original theory to cover an extension of the time base needed for proper handling of efficient distributed simulation. And in keeping with the main theme, it gives an in-depth treatment of hybrid discrete-continuous model representation and simulation.

Several books have, and are scheduled to, come out in the last decade that draw upon the Discrete Event Systems Specification (DEV) theory and applications in one form or another [2-5]. However, as claimed in its jacket, Nutaro’s book occupies a unique position on this bookshelf. Along with the DEV theory, Nutaro presents the actual code of his adevs[6] simulator and explains it in terms of the theory that he develops. This gives you dual perspectives to understand both theory and implementation, either of which could readily become unfathomable standing alone. Along the way, the author also discusses implementation issues in C++ that arise in this context and how the adevs implementation handles them. A major payoff is that, not only will simulation software developers be able to confidently use the open source adevs code, but they also will be in a good position to extend and apply this, as well as other, DEV implementations.

Another beneficial feature is that this book provides examples and applications to illustrate the theory and implementation. A small number of themes are carried through from beginning to end. The main one relates to modeling and simulating a robotic vehicle which is developed in successive stages synchronized to the theoretical progression. A hobbyist’s model tank,

1 Wiley (December 2010, 347 pages)
outfitted with microprocessor control and motor actuator, serves as a down-to-earth system that can be measured, modeled, designed, and simulated with the fidelity needed to get predictions in reasonable agreement with reality. In contrast to most applications, the interested reader need not take the author’s data or word for it; one can simply purchase the parts for a few dollars and build both the real tank and its simulation for oneself.

**Conclusions: Hybrid Simulation and Emerging Technology**

The robotic tank model well illustrates the conjunction of continuous and discrete event dynamics that tomorrow’s engineers will need to master. Learning DEVS and building software for such hybrid simulation will probably never attain anywhere the near the popularity of today’s programming technologies – it is hard to imagine a book called “DEVS for Dummies!” Nevertheless, after the more easily done information technology is completed, there will remain the challenges of linking the self-contained abstractions of programming with the physical reality of systems that must work with them – think of the complexities that smart cars will be bringing as they take over more decision making from drivers and interact in probably unintuitive (emergent) ways on streets and highways. Moreover, increasingly it is being recognized that a nation’s competitive capability in such new technologies will determine whether it continues to enjoy today’s standard of living. Hard as it is, building DEVS-based hybrid simulation will give students a uniquely marketable skill set enabling them to participate in the knowledge-based technology solutions of the rapidly arriving future.

**References**


