Theory of Modeling and Simulation

Integrating Discrete Event and Continuous Complex Dynamic Systems

Second Edition

BERNARD P. ZEIGLER
Electrical and Computer Engineering Department
University of Arizona
Tucson, Arizona

HERBERT PRAEHOFER
Institute of Systems Science
Johannes Kepler University
Linz, Austria

TAG GON KIM
Department of Electrical Engineering
Korea Advanced Institute of Science and Technology
Taejon, Korea

ACADEMIC PRESS

An Imprint of Elsevier

Amsterdam Boston Heidelberg London New York Oxford
Paris San Diego San Francisco Singapore Sydney Tokyo

Preface to the First Edition

A Theory of Modeling and Simulation—Why?

The field of modeling and simulation is as diverse as the concerns of man. Every discipline has developed, or is developing, its own models and its own approach and tools for studying these models. Why then is an overall theory of modeling and simulation necessary?

Why should a college senior or first-year graduate student majoring in biology, aeronautical engineering or business management be introduced to the general theory of modeling and simulation—yet another discipline not specific to his own? The answer relies on the same reasoning that determined that the student should have acquired at least some grounding in mathematics. Nobody questions the role of arithmetic in the sciences, engineering, and management. Arithmetic is all pervasive, yet it is a mathematical discipline having its own axioms and logical structure. Its content is not specific to any other disciplines but is directly applicable to them all. Thus students of biology and engineering are not taught how to add differently—the different training comes in what to add, when to do it, and why.

The practice of modeling and simulation too is all pervasive. However, it has its own concepts of model description, simplification, validation, simulation, and exploration, which are not specific to any particular discipline. These statements would be agreed to by all. Not everyone, however, would say that the concepts named could be isolated and abstracted in a generally useful form.

In response, this book attempts to provide a framework in which the concepts can be sketched in their clear abstract forms; embodying the concepts in concrete modeling situations further serves to illuminate them. Of what use is such a framework for the specializing student, or the one who later will specialize? It can fulfill the following purposes:

1. Introduce the student to the full range of concepts and tools available in the field, rather than just those currently employed in his discipline.
2. Provide a perspective on the modeling activity of his own discipline, permitting the creative employment of insights and approaches stemming from other disciplines.
3. Provide an understanding of why we model and simulate, what can be achieved, and what cannot be achieved.
4. Provide a universal language for communicating the structure and behavior of models to others, regardless of whether they are conversant with the interpretive frameworks of their respective disciplines.

We have been talking about the theory of modeling and simulation relative to the noncomputer sciences. However, the book is also addressed to those who are interested in general and mathematical systems theory or in the more specialized computer and information sciences. Indeed, the framework developed here reflects several related influences:

- General systems theory, with its belief in an underlaying and exploitable unity in all systems, diverse though they may seem.
- Mathematical systems theory, with its attempts to formalize the concepts of system structure and behavior.
- Automata theory, with its formal, logical, and algebraic analysis of the behavior and interrelations of computer-related models.

In this context this book marshals the available approaches and formalisms in order to deal with the particular issues raised by modeling and simulation. Thus the theory presented should introduce the student of systems and computer science to the many open problems in the area needing conceptual and/or mathematical development, as well as furnish a basis for attacking them.

For the veteran practitioner of modeling and simulation, the book will fill an important gap in the literature between the material that is very abstract and that which is very applied. The reader who has had little formal mathematics may find the going rough. Still, my hope is that he will be led to more clearly perceive the issues faced daily in his modeling efforts.

B.P.Z.
Ann Arbor, 1976

Preface to the Second Edition

This is the second edition of *Theory of Modeling and Simulation* originally published by Wiley Interscience in 1976 and reissued by Krieger Publishers in 1984. The first edition made the case that a theory was necessary to help bring some coherence and unity to the ubiquitous field of modeling and simulation. Although nearly a quarter of a century later there have been many advances in the field, we believe that a widely accepted framework and theoretical foundation are even more necessary today. Modeling and simulation lore is still fragmented across the disciplines making it difficult to share in the advances, reuse other discipline's ideas, and work collaboratively in multidisciplinary teams. As a consequence of the growing specialization of knowledge, there is even more fragmentation in the field now than ever. The need for "knowledge workers" who can synthesize disciplinary fragments into cohesive wholes is increasingly recognized. Modeling and simulation—as a generic, non-discipline-specific, set of activities—can provide a framework of concepts and tools for such knowledge work.

In the years since the first edition, there has been much significant progress in modeling and simulation, but the progress has not been uniform across the board. Generally, model building and simulation execution have been made easier and faster by riding piggyback on the technology advances in software (e.g., object-oriented programming) and hardware (e.g., faster processors). However, hard, fundamental issues such as model credibility (e.g., validation, verification, and model family consistency) and interoperation (e.g., repositories, reuse of components, and resolution matching) have received a lot less attention. But these issues are now moving to the front and center under the impetus of the High Level Architecture (HLA) standard mandated by the United States Department of Defense for all its contractors and agencies.

In this edition, two major contributors to the theory of modeling and simulation join with the original author to completely revise the original text. As suggested by its subtitle, the current book concentrates on the integration of the continuous and discrete paradigms for modeling and simulation. A second major theme is that of distributed simulation and its potential to support the coexistence of multiple formalisms in multiple model components.

Although the material is mostly new, the presentation format remains the same. There are three major sections. Part I introduces a framework for modeling and simulation and the primary continuous and discrete approaches to making models and simulating them on computers. This part offers a unified view of the field that most books lack and, written in an informal manner, it can be used as instructional material for undergraduate and graduate courses.

Part II revisits the introductory material but with a rigorous, multi-layered systems theoretic basis. It then goes on to provide an in-depth account of models as systems specifications, the major systems specification formalisms and their integration, and simulators for such formalisms, in sequential, parallel, and distributed forms.

The fundamental role of systems mosphisms is taken up in Part III—any claim relating systems, models and simulators to each other ultimately must be phrased with an equivalence or morphism of such kinds. Both perfect and approximate morphisms are discussed and applied to model abstraction and system representation. Especially, in the latter vein, we focus on the ability of the DEVS (Discrete Event System Specification) formalism to represent arbitrary systems, including those specified in other discrete event and continuous formalisms. The importance of this discussion derives from two sources—the burgeoning use of discrete event approaches in high technology design (e.g., manufacturing control systems, communication, computers) and the HLA-stimulated growth of distributed simulation, for which discrete events match the discreteness of message exchange.

Part IV continues with the theme of DEVS-based modeling and simulation as a foundation for a high-technology systems design methodology. We include integration with other formalisms for analysis and the system entity structure/model base concepts for investigating design alternatives and reusing good designs. Thoughts on future support of collaborative modeling and simulation close the book.

Although the book is primarily intended as a reference, its structure makes it appropriate for use as a textbook in graduate courses on modeling and simulation. As a textbook, the book affords the advantage of providing an open systems view that mitigates the closed-box trust-on-faith approach of many commercial domain-specific simulation packages. If nothing else, the student will have a more sophisticated skepticism about the model reliability and simulator correctness inside such boxes. For hands-on experience, the book needs to be supplemented with an instructional modeling and simulation environment such as DEVSJAVA (available from the Web site: arizona.edu). Other books on statistical aspects of simulation and application to particular domains should be part of the background as well.

We suggest that Part IV might be a good place to start reading, or teaching, since most of the concepts developed earlier in the book are put into use in the last chapters. In this strategy, the learner soon realizes that new concepts are needed to achieve successful designs and is motivated to fill in the gaps by turning to the chapters that supply the requisite knowledge. More likely, a good teacher will guide the student back and forth between the later and earlier material.

Space limitations have prevented us from including all the material in the first edition. The decision on what to leave out was based on relevance to the current theme, whether significant progress had been made in an area, and whether this could be reduced to the requirements of a book. Thus, for example, a major omission is the discussion of structural inference in Chapters 14 and 15 of the original. We hope that a next revision would be able to include much more on developments in these important directions.

B.P.Z.
H.P.
T.G.K.

Acknowledgments

The authors have, when combined, way too many mentors, colleagues, students, and sponsors to acknowledge individually. Our mentors have provided the right word at the right time, and their colleagues have collaborated to develop concepts and software that are beyond the individual capacities to achieve. Our graduate students have struggled with the theories and the recalcitrant software tools to implement prototypes that demonstrated key principles. And of course our sponsors have made it all possible, often with only faith as collateral. The relationship of the authors to each other illustrates the student-mediated intergenerational transmission process. One of the book's authors, Tag Gon Kim, now a full professor at KAIST (Korea Advanced Institute of Science and Technology), obtained his Ph.D. degree under supervision of the senior author (BPZ) at the University of Arizona. Herbert Praehofer, the third author, is now a member of the Systems Science Institute at the University of Linz, Austria, where he also obtained his Ph.D. degree under Franz Pichler, a colleague of Zeigler's. HP especially thanks Prof. Pichler for initiating his interest in systems theory and for the continuing promotion of his work.

Starting near the beginning, Tunçer Ören of Turkey and Canada, and Maurice Elzas of the Netherlands, initiated a series of meetings that were critical to the development of modeling and simulation methodology. Along the way Art Burks, John Holland, George Klir, Ghi Vansteenkiste, Richard Nance, Francois Cellier, Larry Ho, Andrew Sage, Alex Sisti, Paul Davis, and Judith Dahmann provided encouragement at critical turning points. Colleagues Roger Weinberg, Kyetu Ho Park, Paul Fishwick, George Ball, and Steve Hall played important roles in theory and software developments. Of all the students that have contributed important advances, Jerzy Rozenblit, Hessam Sarjoughian, and Adelinde Uhrmacher, now noted researchers in their own right, deserve special mention. Among the many other graduates that dragged us, sometimes hesitantly, in new directions, space allows us to mention David Belogus, Art Concepcion, Doo Kwan Baik, Joon Sung Hong, Ki Hyung Kim, Hyu Chan Park, Kyung Pyo Hong, Yound Jae Kim, Wan Bok Lee, Yi Won Kwon, Alex Chow, Maria Vasconselos, Doo Hwan Kim, Yoonkeon Moon, Gernot Reisinger, Sung-Do Chi, Jerry Couretas, and Hyup Jae Cho. From his varied roles in the United States Department of Defense, Dell Lunceford was the mover and shaker behind the Advanced Simulation Technology Thrust, a DARPA initiative that funded advanced work on DEVS, some of which is described in Chapter 16.

Finally, we want to thank our parents, wives, and children for their love, patience, and understanding.

B.P.Z.
H.P.
T.G.K.

Contents

I

Basics