# Editorial for Principles of Advanced Discrete Simulation

Modeling and simulation has witnessed a surge of advanced methods and tools to improve the tasks of the simulation users, ranging from advanced methodologies, software development methods, middleware, visualization tools, analytics, Cloud computing, standards, and others. Within this context, the new ACM SIGSIM Conference on Principles of Advanced Discrete Simulation (SIGSIM-PADS) has allowed sharing and discussing new methods, tools and applications in this field. The new SIGSIM-PADS conference was rebuilt upon 26 years of history and the reputation for high-quality papers of the traditional PADS Workshop (Workshop on Principles of Advanced and Distributed Simulation). The objective of this new conference is now the intersection of computer science and modeling and simulation.

The inaugural conference attracted 75 full paper submissions from 27 countries around the world. From the 29 papers accepted, the top 10 were invited to submit extended versions. There was also a call for papers for articles related to this field of research. The numerous articles (some of which were new, and others were extended versions of the original SIGSIM-PADS articles) were peer reviewed again by at least three reviewers, making the quality of the final submissions very high. The topics were varied, ranging from automation in parallel simulation in multicore architectures, generation of simulation models, management of activity in simulation, heterogeneous parallel simulation, and theory and emergent behavior. We also received articles on agent-based simulation, simulation with hardware-in-the-loop, and varied applications including networking and communications, logistics, and biology.

The articles presented in this issue can be categorized, based on their research area, into the following areas: modeling methodologies, parallel and distributed simulation, and network simulation and emulation.

## Modeling Methodologies

The article “Formalization of Weak Emergence in Multiagent Systems” by Claudia Szabo, and Yong Meng Teo introduces a new method to represent emergence in simulation models. The authors present new techniques to extract emergence information (flocking, hubs, etc.) from agent-based models. They present a generic method that does not require previous knowledge about the model itself and can be adapted to various applications. They introduce a grammar-based solution that can automate the whole process.

“Automatic Runtime Adaptation for Component-Based Simulation Algorithms” by Tobias Helms, Roland Ewald, Stefan Rybacki, and Adelinde M. Uhrmacher presents advanced algorithms aimed to adapting the simulators dynamically at runtime. The objective of such adaptation is to improve the overall performance. The authors present a method that uses reinforcement learning to explore the algorithms’ performance during a simulation run and adapt the runtime execution based on current data. They used a PDEVS simulator and an ML-rules simulator as a case study to show the feasibility of the approach.

The article “Static Analysis Techniques for Semi-Automatic Synthesis of Message Passing Software Skeletons” by Jason Dagit, Deli Zhang, Gilbert Hendry, and Damian Dechev focuses on a new methodology to design high-performance computing

© 2015 ACM 1049-3301/2015/12-ART1e $15.00
DOI: http://dx.doi.org/10.1145/2845147

architectures. They propose the use of "program skeletons" to analyze large-scale applications, providing a mechanism to analyze programs and to extract information to build program skeletons. They use the SST/macro simulator to build a case study that demonstrates the quality of the extraction process, in which they compare communication trances and performance speedups.

### Parallel and Distributed Simulation

The article "Adaptive Resource Provisioning Mechanism in VEEs for Improving Performance of HLA-Based Simulations" by Zeng Xiang Li, Wentong Cai, Stephen John Turner, Xiaorong Li, Nguyen Binh Duong, Ngee Ann, and Rick Siow Mong Goh focuses on advanced mechanisms for distributed simulation using the HLA standard. They attack the performance problems caused by execution rollbacks in imbalanced HLA federations. They used Cloud and data centers with virtual execution environments (VEEs) to speed up execution times. They test their solution with a real-world model, showing the feasibility of their approach.

"Efficient Parallel Discrete Event Simulation on Cloud/Virtual Machine Platforms" by Srikanth B. Yoginath, and Kalyan S. Perumalla uses Cloud computing methods combined with virtual machines to improve the performance of parallel simulation software. They conduct a series of experiments focused on finding the main characteristics of parallel simulation on cloud computing services using virtual machines. They conducted various experiments on the Amazon EC2 cloud and on a host virtual machine. They design and implement a new scheduler to deal with the parallel execution of the models, showing significant performance gains.

### Network Simulation and Emulation

The article "Symbiotic Network Simulation and Emulation" by Miguel A. Erazo, Rong Rong, And Jason Liu, presents the results obtained when using a testbed for network emulation. They designed and implemented a testbed that mixes a large-scale network and a high-fidelity emulator. The idea is to combine the emulated network and the real hardware/software configurations. They analyze the symbiotic relationship between both environments, trying to reproduce real-world traffic while achieving execution speedups. They present a real world implementation and different case studies to show their new approach to solving this complex problem.

The article "A Methodology to Model the Execution of Communication Software for Accurate Network Simulation" by Stein Kristiansen, Thomas Plagemann, and Vera Goebel focuses on the performance on protocol execution in the nodes of a distributed system. The authors extract information from real devices and propose using that information to extend network simulators, which, by including the real-world hardware/software overhead information, would improve their accuracy. Their models are high level and can be used to describe varied communication software. They present a case study using an extension to the Ns-3 simulator in which they use their methodology to model two different kinds of smartphones, validating the results with real-world data.

"Parallel Simulation and Virtual-Machine-Based Emulation of Software-Defined Networks" by Dong Jin, and David M. Nicol introduces a parallel simulation environment, combined with emulation, in order to study software-defined networks (SDNs). They use OpenFlow to build a combination between network emulation and parallel simulation, and they use the environment to conduct different experiments. They mix a hybrid network testbed and a parallel simulator working together. Their test case is a data center network, in which they use the proposed architecture to improve performance and quality of service by introducing an asynchronous algorithm for passive SDN controllers and a two-tier architecture for the controllers.

The variety of the articles and the high quality of the articles presented will give the author an idea of the most recent advances in this field and will allow the reader to get insight on the top research being conducted in this area.

We extend our gratitude to the numerous reviewers of the original conference papers and the new extended versions submitted to *ACM TOMACS*. Finally, we would like to thank the valuable work by all the authors for their hard work in preparing this issue. Their efforts, diligence, and thoroughness are gratefully acknowledged.

Gabriel A. Wainer

Professor
Dept. of Systems and Computer Engineering
Carleton University Centre for Visualization and Simulation (V-Sim)
1125 Colonel By Drive. 3216 V-Sim
Ottawa, ON. K1S 5B6, Canada
gwainer@sce.carleton.ca
http://www.sce.carleton.ca/faculty/wainer.html