Switching to High Gear: Opportunities for Grand-scale Real-time Parallel Simulations

Kalyan S. Perumalla
Oak Ridge National Laboratory
Oak Ridge, Tennessee, USA
perumallaks@ornl.gov

Abstract

The recent emergence of dramatically large computational power, spanning desktops with multi-core processors and multiple graphics cards to supercomputers with 10^5 processor cores, has suddenly resulted in simulation-based solutions trailing behind in the ability to fully tap the new computational capacity. Here, we motivate the need for switching the parallel simulation research to a higher gear to exploit the new, immense levels of computational power. The potential for grand-scale real-time solutions is illustrated using preliminary results from prototypes in four example application areas: (a) state- or regional-scale vehicular mobility modeling, (b) very large-scale epidemic modeling, (c) modeling the propagation of wireless network signals in very large, cluttered terrains, and, (d) country- or world-scale social behavioral modeling. We believe the stage is perfectly poised for the parallel/distributed simulation community to envision and formulate similar grand-scale, real-time simulation-based solutions in many application areas.

1. Overview

New and exciting developments in computational hardware and software platforms are now offering the parallel/distributed, real-time simulation community unprecedented opportunities. The opportunities may be classified as belonging to two different kinds. In the first kind, opportunities arise in developing natural extensions of current techniques, albeit applied to more potent platforms – new algorithmic methods can be invented, designed, developed, and customized for the emerging large-scale platforms. The second class of opportunities is less obvious – it is the set of opportunities available to revisit *solution approaches* themselves in simulation-based applications. In this latter set, more importantly than in the first kind, and perhaps more urgently, there is a need to take a step back to expand the view of solution approaches, to allow the community to contemplate about and create formulations of new simulation-based solutions that were hitherto fore unimaginable due to computational limitations.

The differential between the computing needs of extant simulation-based solutions and the actual computing capacity offered by the new hardware has, in our view, now become very wide, and is only poised to widen even more in the next few years. For example, while the largest parallel computers now already contain 10^5 processor cores, common applications of commercial discrete event simulators only execute at the level of 10^0-10^2 cores. Very few simulations are designed from the outset to provide increasing value with increases in offered computing power, thus making it hard to envision the use of new levels of computing capacity as natural extensions of current simulation systems.

One cannot but pause to wonder if our current simulations, in fact, happen to be optimally designed to deliver the best value within the small computing capacities of yesteryears, or if they just happen to be under-designed in order to operate within the confines of small computing power constraints. For example, it is unclear if simulating the complex network of queues in a global supply chain can be performed to high degrees of accuracy using a handful of processor cores, or if there are new, domain-relevant questions that can be *posed for the first time* and answered with better agility and accuracy using greatly many more processors.

An analogy was made in the lighter vein by David Nicol in a keynote in 1997, that parallel simulation research "has been tending to scratch where it doesn't itch" [25, 31]. Several years later, perhaps now it is time to evaluate the status differently, by instead exploring if we have been tolerating any unreachable itches on the back for the lack of a sufficiently long scratching stick.

Either way, the moment has probably arrived to afford the community the luxury of contemplating new solutions in terms of new computing power that is literally two or three *orders of magnitude* larger or faster than the computing capacity demanded by the average simulation-based solutions of today. With respect to real-time operation, at a minimum, the new computing magnitudes may enable many existing, taxing simulations to become readily usable in real-time solutions. In addition, simulations that were previously infeasible or unimaginable may now be attempted, or even perhaps sped up to the levels of real-time operation.

The rest of the document is organized as follows. In Section 2, a few relevant trends in computing hardware and software platforms are outlined. In Section 3, the notions of grand-scale solution approaches are illustrated with four example application areas, followed by a summary of future scope in Section 0.

2. A Few Relevant Computing Trends

It is well-known that computing speeds have been steadily increasing over the past few decades. In a related fashion, the amount of computing capacity has also been increasing. By computing capacity, we mean the total amount of computing power available to a user within a single seamless computing installation.

In terms of computing capacity, a relatively more dramatic growth seems to have taken place only in the past 5-6 years. While the largest supercomputing installations in the years 2002-03 contained 10^3 processors, that number has increased by full two orders of magnitude to 10^5 processor cores by the years 2008-09, and the number is scheduled to further increase by another order of magnitude to the tune of 10^6 processor cores within one or two years [1]. Cloud computing [10], which is another approach to providing a large number of loosely connected computing resources as a single installation, also is experiencing dramatic growth. Accelerators are another segment offering rapid growth. For example, clusters of graphical processing unit (GPU)-based systems tout teraflop systems “under the desk” [18, 29, 30, 38].

In terms of middleware, frameworks and programming languages, existing systems are being effectively scaled, while newer environments are also being explored and designed. Implementations of the Message Passing Interface (MPI) [2] have been able to keep up with the increasing capacity; room for improvement is being pursued in the next MPI-3 standard, with extensions such as non-blocking collectives and fault tolerance support. New languages such as Chapel, X10 and Fortress are being developed with the goal of combining overall software cycle productivity with traditional parallel performance considerations. Unified Parallel C and Cilk are representative of other related scaling efforts on the language and runtime side. The emergence of the Common Unified Device Architecture (CUDA) [29] as a new programming interface has greatly impacted the use of GPUs for general purpose computing, and newer languages such as the Open Computing Language (OpenCL) [24] are being developed to provide even more seamless programmability across heterogeneous parallel computing elements within a single system. Additionally, various combinations of the aforementioned systems can be used to suit a specific hardware platform. For example, parallel programming over a cluster of GPUs may be performed by using MPI for inter-node communication and CUDA for computation on the GPUs [3].

In the parallel simulation community, there have been scaling efforts to exploit the increasing computing power. Frameworks for easily parallelizable approaches such as Monte Carlo simulation are obviously readily scalable. For the more demanding scaling challenge, namely, to be able to utilize the full computing resources in the execution of a single simulation run, systems such as ROSS [5, 20] (and μsik [33, 34], which is one of our engines) are being scaled in an attempt to keep up with the increasing computing capacity.

3. Illustrations

We shall now illustrate the immense increases in application-level capabilities that can be potentially enabled by the new computing power. Examples are used from four different applications: (a) regional-scale mobility models on GPUs, executing semi-aggregate models of movement of millions of vehicles on road networks with millions of network links (b) epidemic disease propagation models executed on large supercomputing platforms, to simulate highly detailed, individual-level reaction and diffusion processes in scenarios with 10^6-10^9 individuals, (c) wireless signal propagation models executed on 10^4-core platforms, for highly detailed signal strength estimations in wireless networks operated in expansive terrains with high geometrical complexity, and (d) social behavioral models executed on GPUs or supercomputing platforms, to model the inter-related behavioral evolution of 10^6-10^{10} entities including individuals, processes, and events.

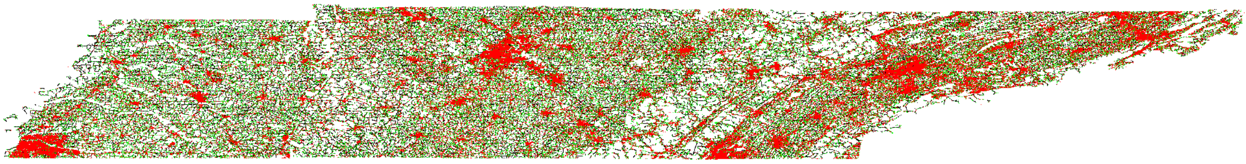

Figure 1: An initial loading of the Tennessee state road network. Uncongested cells are shown in green, congested cells in red.

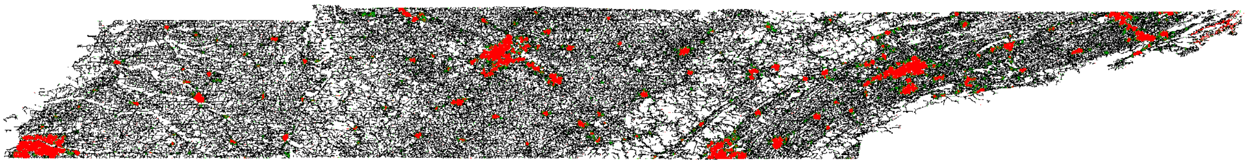

Figure 2: A snapshot in a mobility simulation on the Tennessee state network. Vacated streets are black.

3.1. Regional-scale Mobility

Simulations for applications such as traffic management and emergency planning [7, 17] have traditionally used small-scale scenarios simulated on desktop-based systems. When larger geographical regions are considered in such applications, simulations become highly computationally intensive. In newer solutions, GPU hardware could be tapped to simulate much larger, unprecedented scenarios in real-time on desktop-based systems. Network scenarios that are two to three orders of magnitude larger could potentially be supported. The question of how to most effectively utilize the new scale in existing applications is, in our view, beside the immediate point of interest to the simulation community. Instead, the more pertinent view is the fact that simulation research holds the responsibility to first enable new capabilities. It is then up to the domain users or customers to expand their own views of what is possible, and to envision entirely new applications to build on the new capabilities.

In that vein, we recently examined the challenge of grand-scale scenarios in real-time simulation of mobility – namely, in finding ways to simulate road networks at the scale of entire states or multiple states together (e.g., southeastern US). Such a challenge entails taking the current capabilities of 10^3-10^4 links or intersections to newly supported levels of 10^6-10^7, and, if possible, enabling real-time or faster execution for first-order metrics (e.g., rough estimates of evacuation time for millions of vehicles).

An early version of our prototype implementation is currently operational [36]. Preliminary results show, for the first time ever, the scalability to a field of over 2 million network nodes, 5 million network links, and 20 million represented vehicles. Mobility can be described in a generalized field-based model view. In evacuation simulations, for example, arbitrary fields can be defined to represent any evacuation control scheme. Execution of our prototype implementation shows that results from our system are achieved in real-time, which is significantly faster than any existing vehicular mobility simulator. Simultaneously, the capability with respect to network size is significantly larger, from tens of thousands of nodes of extant systems to millions of nodes in our new system.

Figure 1 shows the initial state of a simulation scenario for mobility over the road network of the state of Tennessee. This represents a geographical area of 140,000 km^2, with 0.58 million intersections and 1.34 million links. Figure 2 shows a snapshot of the simulation taken after a few hours into the evacuation scenario. Figure 3 shows a snapshot of a simulation with the Louisiana state network, which has 0.41 million nodes and 0.99 million links. The largest simulation test was performed with the Texas state road network containing 2.07 million nodes and 5.12 million links [36].

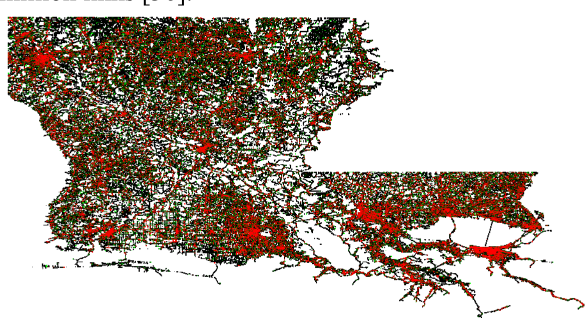

Figure 3: A snapshot in a mobility simulation on the Louisiana state road network. Congested cells are shown in red.

The net effect of the results is that, for the first time ever, state-level or region-level scenarios can be simulated *and visualized* in real-time on desktop platforms. For the first time, it is now possible to *contemplate* grand-scale solutions in this domain (e.g., agile evacuation planning, or dynamically resource-optimized, automated decision-making, for very large geographical regions).

3.2. Large-scale Epidemics

Another application area that affords the potential for scale is the detailed simulation of epidemiological phenomena [4, 12, 14, 16, 19, 21, 22, 32, 39], including details of mutations, reactions, interactions, diffusion, prevention, mitigation, resource limitations, and so on. The spectra of fidelity and scalability in this area are very wide. Our interests are pursuing the limits in the most computationally challenging scales, and verifying or demonstrating the feasibility of simulating at unprecedented scales. As with the previous example, we relegate the contemplation of the most effective use of simulation advancements to domain experts to whom we present radically new computational possibilities.

In the epidemics context, the computational interest lies is dramatically extending the limits on the number of individuals simulated, the complexity of evolution of the infection/strains, the dynamics of diffusion of individuals or carriers, and the feedback effects among all these components.

Towards achieving a proof of feasibility and the potential, we implemented close equivalents of the computational epidemiology models published lately [4]. Our implementation was built using our µsik engine, with novel "inverse models" to aid in reverse computing-based rollback (to be soon reported elsewhere in greater detail), and was subjected to initial scaling experiments. Representative performance results from experiments using weak scaling scenarios are shown in Figure 4 and Figure 5, with 1000 individuals in each interaction cluster, 10 interaction clusters per processor core, and randomized movement of individuals across groups, using different distributions for intra- and inter-core group travel time. Figure 4 shows the amortized average cost of processing each event in the epidemic propagation. Since it represents weak scaling, the flatter the curve the better the efficiency of the parallel execution. Figure 5 plots the run time with increasing number of cores. The results, plotted up to 64K cores, represent the largest number of processors successfully used to date, in discrete event simulation of epidemic propagation. Although much additional analysis and improvement are possible for the performance (e.g., reducing the event cost on larger number of cores), they nevertheless are sufficient to indicate the potential to conceive and execute very large scale scenarios. The number of individuals instantiated in the largest case was over 655 million. Since individual's event is executed within the sub-millisecond range, the entire execution even in the largest case is much faster than real-time.

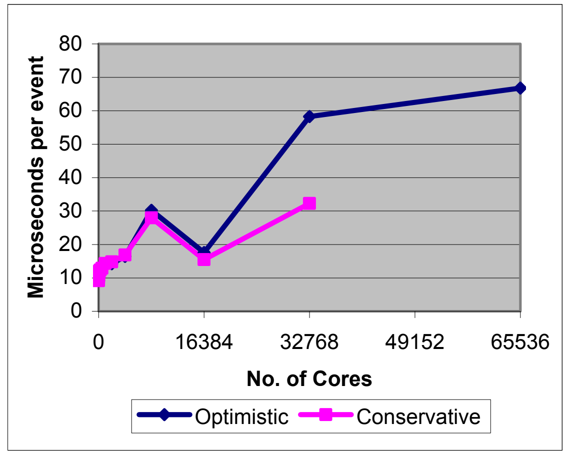

Figure 4: Event processing cost for epidemiological simulation model (weak scaling).

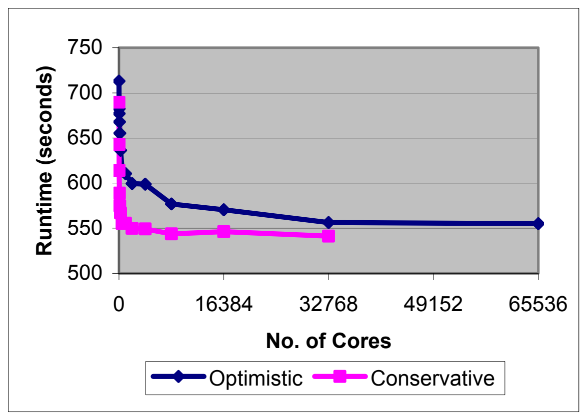

Figure 5: Execution time for epidemiological simulation model (weak scaling).

The main point we make from the initial exercise results is that new, grand-scale solutions could be potentially enabled by the new scale of simulation capabilities. Real-time approximations of ground truth can be conceivably maintained by detailed simulations (e.g., based on daily updates from pharmaceutical drug purchase data, traffic intensity information, actual school/public place closure information, etc.). Perhaps it might be desirable to someday simulate at the level of every individual in the world. Such solution approaches, which were beyond imagination without the possibility of very detailed large-scale, real-time simulation, can now be *contemplated*.

3.3. Wide-terrain Wireless Signal Propagation

The next illustration of grand-scale solution scenario is another simulation application, namely, simulation-based estimation of wireless signal strength [5, 6, 41]. The signal strength estimation problem finds application in many military radio communication design and analysis problems, as well as in simulating and designing industrial wireless systems [11]. Scenarios of interest may span very large geographical domains, and may include irregular features like buildings, trees and other foliage. A recent alternative for such applications is a new (sequential) model based on discrete event formulation [28], which resolves several shortcomings of previous approaches such as finite-difference time domain [8] and ray tracing [40]. It adopts a transmission line matrix (TLM) method which uses equivalent electrical networks to solve partial differential equations that govern the underlying physical phenomenon. Validation studies [28] of the sequential TLM-based discrete event model have shown that its runtime is more efficient than conventional FDTD or ray-tracing methods while preserving desirable accuracy levels of radio signal strength estimates. We have developed highly scalable parallel implementations of this new approach, using optimistic execution of the TLM model with reverse computing-based rollback. The simulation is developed as a µsik application. Results show scalability of our discrete event simulation to tens of thousands of processors [41]. In the largest case, 17,576 cores are used to simulate a terrain volume of size 130×130×130 grid cells. A reduction in runtime by a factor of over 1200× was achieved in this scenario for signal estimation.

The scalability and decrease in runtime in this application enables one to envision use of highly detailed models down to the physical layer in real-time simulation of complex networked combat scenarios using wireless communication, which was previously unimaginable for real-time use.

3.4. Country-scale Social Behaviors

Recently, an increased amount of interest from the academic, government and defense communities is evident in simulation-based behavioral studies for large-population scenarios (e.g., [9, 13, 42]). In the absence of alternative or better methodologies, there appears to be intense interest in using agent/rule-based approaches to understanding behaviors of real social systems. Simulation-based studies are being applied to the new needs of scale and speed in large-scale scenarios of interest.

To illustrate, the defense and intelligence communities are interested in simulating and understanding social behavioral implications of military alternatives in operations such as pre- or post-conflict campaigns. For understanding the implications, the populations of interest could be either domestic or foreign or both. The scales of interest can range from towns and cities to states and countries as well, although the quantitative nature of analysis is usually progressively relaxed to become more qualitative in nature for larger population bodies. Decisions such as the determination of the amount of foreign aid, or the emphasis on infrastructure (living condition) development in the theater of interest may be based on the expected or afforded level of behavioral benefit (e.g., cooperation or non-hostile disposition) from the subject population. Behavioral modeling and simulation may need to be employed in relatively short order when new theaters of interest are identified, and when planning (e.g., deployments or resource allocations) must be performed relatively quickly. The behavioral analysis problem is exacerbated by the fact that the simulation models may need to be applied to population classes that were not specifically studied in advance. For cost-effectiveness, the same models may be reused for disparate theaters, making it necessary for the models to be sufficiently flexible to be customized and configured for different population characteristics.

Agent-based modeling and simulation is a common method of choice in this area, with parallel and/or distributed execution employed to allow instantiation of large scenarios and/or to execute in reasonable amount of time [15, 23, 26, 27, 35]. Even more computationally-taxing models, such as Connectionist Models [37], may be employed for greater fidelity and improved basis for plausibility. Models such as the connectionist models require large computational power to simulate the detailed neurological reactions and self-driven evolution of cognitive properties inside each individual. When evaluated with populations at the country-level, the total number of states and the number of interactions potentially sum up to 10^6-10^9. Although domain science is yet to consider scenarios at such scale, the simulation community can advance the technology to enable analysis at that scale, demonstrating its feasibility and make it accessible to the applications. The domain experts, who previously have not been accustomed to thinking in terms of the large simulation scenario sizes, may then begin to contemplate new ways of using large-scale executions. With such an outlook, we have performed initial feasibility runs using prototypes executed on multiple platforms.

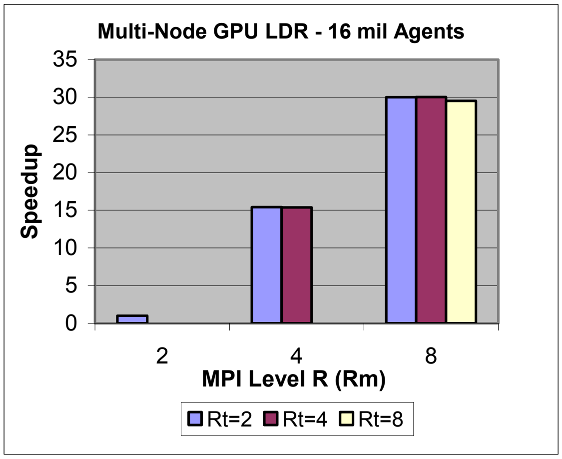

Figure 6: Factor of speed increase with latency hiding technique for agent-based simulation on a cluster of GPUs using MPI and CUDA.

To cater to the desktop-based customers, we investigated the feasibility of using a cluster of commodity GPUs to simulate very large numbers of agents. To deal with the deep memory hierarchies and heterogeneity of computing elements, we developed a latency-hiding scheme to find the optimal communication-computation tradeoff. As a simple example, Figure 6 shows the factor of improvement obtained by latency-hiding schemes when agent-based simulations (in this case, 16 million in a Leadership model) are executed on a cluster of GPUs using a combination of MPI and CUDA. The best performance gain of nearly 30× is observed with a "ghost cell factor" equal to 8 cells on the boundaries along each dimension at the MPI level, and 8 cells at the CUDA thread level.

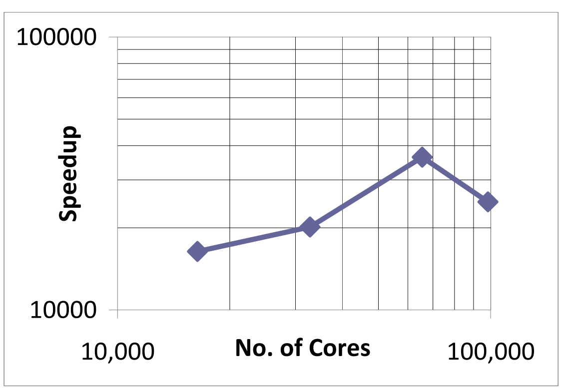

Figure 7: PHOLD weak scaling on Cray XT5

For larger and more complex agents, the PHOLD model can be used as an approximate predictor of runtime performance. Figure 7 shows the speedup obtained by the μsik engine with a moderately loaded version of PHOLD under weak scaling (number of logical processes and message population increased proportionately with the number of processors). Speedup is plotted relative to 16,384 cores as the baseline (i.e., efficiency of 1.0 assumed for 16,384). The main point conveyed by the plot is simply to show evidence of feasibility today of executing parallel discrete event simulations on very large (10^5) numbers of processors with non-trivial speedup.

4. Future Scope

The moment is right to envision new research directions in parallel and distributed simulation. Now is perhaps the perfect window of opportunity for the simulation community to work closely with domain scientists, to consciously expand the outlook from the extant desktop-based solutions to radically larger, unexplored approaches. Considering the dramatic, and rather unexpected, rate of increase in available computing power that has taken place in the very short time period of the past few years, a suspension of disbelief about the potential for grand scale may in fact be warranted briefly, to help the community leapfrog over the confines of the small scale of yesteryears. Grand-scale simulation-based solutions may be formulated using the new, unprecedented computational capabilities, to drive hitherto fore unexplored research directions in large-scale, real-time simulation technologies.

Acknowledgements

This paper has been authored by UT-Battelle, LLC, under contract DE-AC05-00OR22725 with the U.S. Department of Energy. Accordingly, the United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. This research used resources of the National Center for Computational Sciences at Oak Ridge National Laboratory, which is supported by the Office of Science of the Department of Energy under Contract DE-AC05-00OR22725. The author is indebted to Brian Worley, division director, for strong encouragement and support in real-time simulation research. Keynote invitation by Wentong Cai and Stephen Turner are gratefully acknowledged. Parts of the research reported here are joint works with Brandon Aaby, Sudip Seal, Vinod Tipparaju, and Srikanth Yoginath.

References

[1] "Top 500 Supercomputing Sites," http://top500.org.

[2] "Message Passing Interface Forum," http://www.mpi-forum.org/.

[3] B. G. Aaby, K. S. Perumalla, and S. K. Seal, "Parallel Agent-Based Simulations on Clusters of GPUs and Multi-Core Processors," Oak Ridge National Laboratory, Oak Ridge, TN, USA ORNL/TM-2009/183, 2009.

[4] C. L. Barrett, K. R. Bisset, et al., "EpiSimdemics: An Efficient Algorithm for Simulating the Spread of Infectious Disease over Large Realistic Social Networks," in *Proceedings of the 2008 ACM/IEEE Conference on Supercomputing* Austin, Texas: IEEE Press, 2008.

[5] D. W. Bauer, C. D. Carothers, and A. Holder, "Scalable Time Warp on Blue Gene Supercomputers," in *ACM/IEEE/SCS 23rd Workshop on Principles of Advanced and Distributed Simulation* Lake Placid, NY, USA: IEEE Computer Society, 2009.

[6] D. W. Bauer and E. H. Page, "Optimistic Parallel Discrete Event Simulation of the Event-based Transmission Line Matrix Method," in *Winter Simulation Conference* Washington, DC, USA: IEEE Press, 2007.

[7] B. Bhaduri, C. Liu, and O. Franzese, "Oak Ridge Evacuation Modeling System (OREMS): A PC-Based Computer Tool for Emergency Evacuation Planning," in *Symposium on GIS for Transportation*, 2006.

[8] S. D. Bilbao, *Wave and Scaterring Methods for Numerical Simulations*: Wiley, 2004.

[9] P. Brecke, P. Voorheis, et al., "Actionable Capability for Social and Economic Systems (ACSES)," Seedling Project - Defense Advanced Research Projects Agency, Project Report 2008.

[10] R. Buyya, C. S. Yeo, et al., "Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility," *Future Generation Computing Systems,* vol. 25, pp. 599-616, 2009.

[11] J. Chen and S. Hall, "Efficient and Outdoor EM Wave Propagation in a Compact Terrain Database of the Urban Canyon Environment," *Proc. of the Vehicular Technology Conference,* pp. 802-806, 2002.

[12] D. J. Daley and J. Gani, *Epidemic Modelling: An Introduction*. Cambridge, UK: Cambridge University Press, 2001.

[13] DARPA, "US Defense Advanced Research Projects Agency - Technologies for the Applications of Social Computing (TASC)," http://www.darpa.mil/ipto/solicit/baa/RFI-SN-09-20_PIP.pdf.

[14] E. Deelman and B. K. Szymanski, "Simulating Spatially Explicit Problems on High Performance Architectures," *Journal of Parallel and Distributed Computing,* vol. 62, pp. 446-467, 2002.

[15] J. Epstein, "Modeling Civil Violence: An Agent-based Computational Approach," *PNAS,* vol. 99, pp. 7243-7250, 2002.

[16] S. Eubank, H. Guclu, and M. V. Marathe, "Modeling Disease Outbreaks in Realistic Urban Social Networks," *Nature,,* vol. 429, pp. 180-184, 2004.

[17] O. Franzese and L. Han, "A Methodology for the Assessment of Traffic Management Strategies for Large-scale Emergency Evacuations," in *11th Annual Meeting of ITS America*, 2001.

[18] GPGPU, "General Purpose Computation Using Graphics Hardware," http://www.gpgpu.org.

[19] M. E. Halloran, N. M. Ferguson, and e. e. a. S. Eubank, "Modeling Targeted Layered Containment of an Influenza Pandemic in the United States," *Proc. of the National Academy of Sciences,,* vol. 105, pp. 4639-4644, 2008.

[20] A. O. Holder and C. D. Carothers, "Analysis of Time Warp on a 32,768 Processor IBM Blue Gene/L Supercomputer," in *European Modeling and Simulation Symposium* Italy: Liophant, 2008.

[21] J.-L. Hsieh, C.-Y. Huang, et al., "Learning to Build Network-oriented Epidemic Simulation Models in Epidemiology Education," *International Journal of Simulation and Process Modeling,* vol. 5, pp. 31-41, 2009.

[22] C.-Y. Huang, Y.-S. Tsai, et al., "Influences of Resource Limitations and Transmission Costs on Epidemic Simulations and Critical Thresholds in Scale-Free Networks," *SIMULATION,* vol. 85, pp. 205-219, 2009.

[23] M. Hybinette, E. Kraemer, et al., "A Design for a Scalable Agent-based Simulation System using a Distributed Discrete Event Infrastructure," in *Winter Simulation Conference*: IEEE Press, 2006.

[24] Khronos, "Open Computing Language (OpenCL)," http://www.khronos.org/opencl/.

[25] D. Nicol, "Parallel Discrete Event Simulation: So Who Cares?," in *ACM/IEEE/SCS Workshop on Parallel and Distributed Simulation* Lockenhaus, Austria: IEEE Computer Society, 1997.

[26] M. J. North, N. T. Collier, and J. R. Vos, "Experiences Creating Three Implementations of

the Repast Agent Modeling Toolkit," *ACM Transactions on Modeling and Computer Simulation,* vol. 16, pp. 1-25, 2006.

[27] M. J. North and C. M. Macal, *Managing Business Complexity: Discovering Strategic Solutions with Agent-Based Modeling and Simulation*: Oxford University Press, 2007.

[28] J. Nutaro, T. Kuruganti, et al., "An Event Driven, Simplified TLM Method for Predicting Path-loss in Cluttered Environments," *IEEE Trans. on Antennas and Propagation,* vol. 56, pp. 189-198, 2008.

[29] NVIDIA, "Common Unified Device Architecture (CUDA)," http://developer.nvidia.com/cuda.

[30] J. D. Owens, D. Luebke, et al., "A Survey of General-Purpose Computation on Graphics Hardware," in *Eurographics*, 2005, pp. 21-51.

[31] E. H. Page, "Beyond Speedup: PADS, The HLA and Web-based Simulation," in *ACM/IEEE/SCS Workshop on Parallel and Distributed Simulation* Atlanta, GA, USA: IEEE Computer Society, 1999.

[32] J. Parker, "A Flexible, Large-scale, Distributed Agent-based Epidemic Model," in *Winter Simulation Conference* Piscataway, NJ: IEEE Press, 2007.

[33] K. S. Perumalla, "µsik - A Micro-Kernel for Parallel/Distributed Simulation Systems," in *Workshop on Principles of Advanced and Distributed Simulation*, 2005.

[34] K. S. Perumalla, "Scaling Time Warp-based Discrete Event Execution to 10^4 Processors on the Blue Gene Supercomputer," in *International Conference on Computing Frontiers*, 2007, pp. 69-76.

[35] K. S. Perumalla, "Computational Spectrum of Agent Model Simulation," in *Modelling, Simulation and Optimization*: IN-TECH, 2009.

[36] K. S. Perumalla, B. G. Aaby, S. B. Yoginath, and S. K. Seal, "GPU-based Real-Time Execution of Vehicular Mobility Models in Large-Scale Road Network Scenarios," in *ACM/IEEE/SCS 23rd Workshop on Principles of Advanced and Distributed Simulation*: IEEE Computer Society, 2009.

[37] K. S. Perumalla and J. C. Schryver, "A Connectionist Modeling Approach to Rapid Analysis of Emergent Social Cognition Properties in Large-Populations," in *Human Behavior-Computational Modeling and Interoperability Conference* Oak Ridge, Tennessee: IEEE, 2009.

[38] M. Pharr and R. Fernando, *GPU Gems 2: Programming Techniques for High-Performance Graphics and General-Purpose Computation*: Addison Wesley Professional, 2005.

[39] D. M. Rao and A. Chernyakhovsky, "Parallel Simulation of the Global Epidemiology of Avian Influenza," in *Winter Simulation Conference* Miami, Florida: Winter Simulation Conference, 2008.

[40] K. A. Remley, A. Weisshaar, and H. R. Anderson, "A Comparison Study of Ray Tracing and FDTD for Indoor Propagation Modeling," *Proc. of the Vehicular Technology Conference,* pp. 865-869, 1998.

[41] S. K. Seal and K. S. Perumalla, "Scalable Parallel Execution of an Event-based Radio Signal Propagation Model for Cluttered 3D Terrains," in *International Conference on Parallel Processing* Vienna, Austria, 2009.

[42] J. Whitmeyer, "Learning Theories for Loyalty-based Leadership Model," 2007.