A Review of Simulation of Telecommunication Networks: Simulators, Classification, Comparison, Methodologies, and Recommendations

Nurul I. Sarkar, Senior Member, IEEE and Syafnidar A. Halim

Abstract—Simulation methodology has become popular among computer and telecommunication network researchers and developers worldwide. This popularity is due to the availability of various sophisticated and powerful simulation packages, and also because of the flexibility in model construction and validation offered by simulation. For selecting an appropriate network simulator for a simulation task, it is important to have good knowledge of the simulation tools available, along with their strengths and weaknesses. It is also important to ensure that the results generated by the simulators are valid and credible. The objective of this paper is to survey, classify, and compare telecommunication network simulators to aid researchers in selecting the most appropriate simulation tool. We compare the network simulators based on type, deployment mode, network impairments and protocol supported. We discuss simulator evaluation methodologies and techniques, and provide guidelines for best practice in network simulation.

Index Terms— Network simulator, simulation methodology, parallel simulation.

I. Introduction

Network simulation methodology is often used to verify analytical models, generalize the measurement results, evaluate the performance of new protocols that are being developed, as well as to compare the existing protocols. However, there may be a potential problem when using simulation in testing protocols because the results generated by a simulator may not be necessarily accurate or representative. To overcome this problem, it is important for network researchers and developers to use a credible simulation tool which is easy to use; more flexible in model development, modification and validation; and incorporates appropriate analysis of simulation output data, pseudo-random number generators, and statistical accuracy of the simulation results. To select a credible simulator for a simulation task, it is also important to have good knowledge of the available simulation tools, along with their relative strengths and weaknesses. These aspects of credible simulation studies are recommended by leading simulation researchers [1-3].

The use of discrete event simulation packages as an aid to modeling and performance evaluation of computer and telecommunication networks has grown in recent years [4-6]. This popularity results from the availability of sophisticated simulators and low cost powerful personal computers (PCs).

A detailed discussion of simulation methodology, in general, can be found in [2, 7]. More specifically, Pawlikowski [8] in a comprehensive survey of problems and solutions suited for steady-state simulation mentioned the relevance of simulation technique for modeling telecommunication networks.

In this paper we survey existing network simulators highlighting their strengths and weaknesses. We classify and compare popular simulators based on type and deployment mode along with network impairments and protocol supported. The simulation methodologies, evaluation techniques and credibility of simulation studies are discussed. Telecommunication network researchers and developers can use the results of this study in selecting the most appropriate simulator.

The rest of the paper is organized as follows. Section II surveys popular network simulators highlighting their strengths and weaknesses. In Section III, we describe simulation methodologies and techniques including credibility of simulation studies. Section IV provides recommendations for best practice in network simulation, and a brief conclusion in Section V concludes the paper.

II. A Survey of Existing Network Simulators

While various simulators exist for building a variety of network models, we compare 10 popular network simulators highlighting their strengths and weaknesses. These simulators were selected based on their popularity, published results, and interesting characteristics and features.

A. Commercial network simulator

i) OPNET: Optimized Network Engineering Tool (OPNET) is a discrete event, object-oriented, general purpose network simulator. It provides a comprehensive development environment for the specification, simulation and performance analysis of computer and data communication networks.

OPNET is a commercial network simulation package which is available for supporting both the teaching and research in educational institutions under the OPNET university academic program [9]. OPNET has several modules and tools, including

OPNET modeler, planner, model library, and analysis tools [10]. It is widely used in the network industries for performance modeling and evaluation of local and wide-area networks.

The main strengths of OPNET include a comprehensive model library, modular model development, high level of modeling detail, user-friendly GUI, and customizable presentation of simulation results. However, OPNET is a very expensive package (license maintenance fees are also high), and its parameter categorization is not very transparent.

ii) QualNet Developer: QualNet Developer (‘QualNet’) is a distributed and parallel network simulator that can be used for modeling and simulation of large networks with heavy traffic [11]. The QualNet consists of QualNet scenario designer, QualNet animator (visualization and analysis tool), QualNet protocol designer (protocol skeleton tool), QualNet analyzer (real time statistical tool), and QualNet packet tracer (visualization and debugging tool). QualNet is a commercial version of the open source simulator called GloMoSim.

The main strength of QualNet is that it supports thousands of nodes and run on a variety of machines and operating systems. It has a comprehensive network relevant parameter sets and allows verification of results through by inspection of code and configuration files. However, QualNet does not have any predefined model constructs.

iii) NetSim: NetSim is available both commercial and academic versions, and can be used for modeling and simulation of various network protocols, including WLANs, Ethernet, TCP/IP, and asynchronous transfer mode (ATM) switches [12]. NetSim allows a detailed performance study of Ethernet networks, including wireless Ethernet. The effect of relative positioning of stations on network performance, a realistic signal propagation modeling, the transmission of deferral mechanisms, and the collision handling and detection processes can also be investigated [10].

The main strength of NetSim is that the package can be run on a variety of operating systems. However, the use of NetSim is limited to academic environments only.

iv) Shunra Virtual Enterprise (Shunra VE) 5.0: Shunra VE is a hardware-based simulation environment having an advantage of high speed than the software-based simulation [13]. The network impairments supported are the latency, bandwidth, jitter, packet loss, bandwidth congestion and utilization [14]. StormCather enables the replay and capture of network activities. StormConsole used as the interface to StormAppliance, creates the network model [13].

The main strength of Shunra VE include hardware-based system, good support, empirical model and uses real-life appliances. However, it is a very expensive package and requires a good network infrastructure for up and running.

B. Open source network simulator

i) Ns-2: Ns-2 is one of the most widely used network simulators in use today. It is an object-oriented discrete-event network simulator originally developed at Lawrence Berkeley Laboratory at the University of California, Berkeley, as part of the Virtual InterNetwork Testbed (VINT) project [15]. It was primarily designed for network research community for simulating routing algorithms, multicast, and TCP/IP protocols. The Monarch project at Carnegie Mellon University has extended the ns-2 with support for node mobility [16]. Ns-2 is written in C++ and uses OTcl as a command and configuration interface.

The main strength of ns-2 is its availability for download on a variety of operating systems at no costs. Authors of research papers often publish ns-2 code that they used, allowing other researchers to build upon their work using the original code. This is particularly useful to academia, specifically Master’s and Doctoral students who are looking for a tool for network modeling and performance evaluation.

The main weakness of ns-2 is the lack of graphical presentations of simulation output data. The raw data must be processed using scripting languages such as ‘awk’ or ‘perl’ to produce data in a suitable format for tools like Xgraph or Gnuplot [15]. Another disadvantage of ns-2 is that it is not a user-friendly package because of its text-based interface, and many student researchers point out that ns-2 has a steep learning curve. A tutorial contributed by Marc Greis [17] and the continuing evolution of ns documentation have improved the situation, but ns-2’s split-programming model remains a barrier to many developers.

ii) GloMoSim: It is a library-based parallel simulator, developed at the University of California, Los Angeles, for mobile wireless networks [18]. It is written in PARSEC (Parallel Simulation Environment for Complex System), which is an extension of C for parallel programming. GloMoSim is a scalable simulator that can be used to support research involving simulation and modeling of large-scale networks with thousands of nodes.

The main strength of GloMoSim is its scalability to support thousands of nodes and executing simulation on multiple machines. Although GloMoSim was designed for both wired and wireless networks, currently it supports wireless networks only.

iii) OMNeT++: It is a modular component-based discrete event simulator [19]. It uses building blocks called modules in the simulator. There are two types of modules used in OMNeT++, namely, simple and compound. Simple modules are used to define algorithms and are active components of OMNeT++ in which events occur and the behavior of the model is defined (generation of events, reaction on events). Compound modules are a collection of simple modules interacting with one another.

The main strengths of OMNeT++ include GUI, object inspectors for zooming into component level and to display the state of each component during simulation, modular architecture and abstraction, configurable, and detailed implementation of modules and protocols. However, OMNeT++ is a bit slow due to its long simulation run and high memory consumption. OMNeT++ is also a bit difficult to use.

iv) P2PRealm: Peer-to-Peer Realm (P2PRealm) is a Java based P2P network simulator that can be used in simulating and optimizing neural networks [20]. It was developed as part of Cheese Factory P2P research project (www.mit.jyu.fi/cheesefactory/index.shtml). P2PRealm has four main components: P2P network, algorithms, input/output interface, and neural network optimization. By using P2PRealm, one can verify P2P networks for a topology management algorithm and then produced an output of a neural network [20].

The main strength of P2PRealm is its ability to optimize neural networks used in P2P networks. However, P2P network is still under development.

v) The Georgia Tech Network Simulator: The Georgia Tech Network Simulator (GTNetS) can be used to develop moderate to large-scale simulation models by using existing network simulation tools [21]. Because of the object-oriented methodology, the model developed under GTNetS can be extended easily to support new networking paradigm. The main strength of GTNetS is that the design of GTNetS closely matches the design of real network hardware and therefore with a little knowledge of networking, the model can be constructed and simulated. However, it is still under ongoing development.

vi) AKAROA: AKAROA is a fully automated simulation tool developed at the University of Canterbury, Christchurch, New Zealand. The main design goal was to run existing simulation programs in multiple replications in parallel (MRIP) scenario. AKAROA accepts an ordinary sequential simulation program and automatically launches the number of simulation engines requested by a user. AKAROA-2 is the latest version of AKAROA, which can be used in teaching in addition to research. More details about AKAROA can be found in [22]. The main strength of AKAROA is its MRIP to run simulation faster. However, AKAROA is a bit difficult to use.

C. Comparison

Table I compares 10 popular network simulators based on selected criteria such as simulator type (i.e. commercial or open source), deployment mode (enterprise, small and large scale), network impairments and protocol supported.

The simulator and the corresponding type are listed in column 1 and 2, respectively. The deployment mode in each of the 10 simulators is shown in column 3. The network impairments and protocol supported by each of the simulator are highlighted in column 4 and 5, respectively.

As shown in Table I, the first four simulators namely, OPNET, QualNet, NetSim, and Shunra VE are commercial simulators and the remaining six are open source (ns-2, GlomoSim, OMNeT++, P2P Realm, GTNetS and AKAROA). While commercial network simulators support a wide range of protocols, those simulators released under open source are more specialized on one specific protocol. However, OMNeT++ offers a dual licensing. The source code is released as open source which is available for download at no costs whereas the commercial version called OMNEST [23].

To get an insight into the simulation tools used in the selected IEEE Journal and Conference published papers, we survey all papers published in the IEEE Transactions on Communications (1071 papers), IEEE/ACM Transactions on Networking (377 papers), and in proceedings of IEEE GLOBECOM (2991 papers), INFOCOM (817 papers), and ICC (3114 papers) between 2007 and 2009. A total of 8370 papers were surveyed. The survey results are summarized in Table II. About 42.8% of 8370 papers surveyed have mentioned that they use ns-2 for network modeling and simulation tasks. About 36.8% of the total papers surveyed have used MATLAB whereas 7.6% used OPNET. The remaining 4.2%, 1.6% and 0.8% of the total papers surveyed have used QualNet, GlomoSim, and OMNet++, respectively. We found that about 6.2% of the papers surveyed did not bother to mention the name of the simulators that they had used. We categorize them as others which also include user written programs.

TABLE I. COMPARISON OF POPULAR NETWORK SIMULATORS

Simulator	Type	Deployment mode	Network impairments	Network protocol supported
OPNET	Commercial /academic	Enterprise	Link models such as bus and point-to-point (P2P), queuing service such as Last-in-First-Out (LIFO), First-in-First-Out (FIFO), priority non-preemptive queuing, round-robin.	ATM, TCP, Fiber distributed data interface (FDDI), IP, Ethernet, Frame Relay, 802.11, and support for wireless.
QualNet	Commercial	Enterprise	Evaluation of various protocols.	Wired and wireless networks; wide-area networks.
NetSim	Commercial /academic	Large-scale	Relative positions of stations on the network, realistic modeling of signal propagation, the transmission deferral mechanisms, collision handling and detection process.	WLAN, Ethernet, TCP/IP, and ATM
Shunra VE	Commercial	Enterprise	Latency, jitter and packet loss, bandwidth congestion and utilization.	Point-to-point, N-Tier, hub and spoke, fully meshed networks.
Ns-2	Open source	Small-scale	Congestion control, transport protocols, queuing and routing algorithms, and multicast.	TCP/IP, Multicast routing, TCP protocols over wired and wireless networks.
GloMoSim	Open source	Large-scale	Evaluation of various wireless network protocols including channel models, transport, and MAC protocols.	Wireless networks.
OMNeT++	Open source	Small-scale	Latency, jitter, and packet losses.	Wireless networks
P2P Realm	Open source	Small-scale	Verify P2P network requirements, topology management algorithm or resource discovery.	Peer to peer (P2P)
GTNetS	Open source	Large-scale	Packet tracing, queuing methods, statistical methods, random number generators.	Point-to-Point, Shared Ethernet, Switched Ethernet, and Wireless links.
AKAROA	Open source	Small-scale	Protocol evaluation.	Wired and wireless networks, Ethernet.

TABLE II. SIMULATORS USED IN THE SELECTED IEEE JOURNAL AND CONFERENCE PAPERS PUBLISHED FROM 2007 TO 2009

Simulator	IEEE Transactions on Communications	IEEE/ACM Transactions on Networking	IEEE GLOBECOM	IEEE INFOCOM	IEEE ICC	Overall (%)
ns-2	14%	57%	45%	39%	59%	42.8
OPNET	6%	4%	8%	3%	17%	7.6
MATLAB	78%	32%	29%	32%	13%	36.8
QualNet	-	1%	5%	12%	3%	4.2
GloMoSim	-	1%	1%	3%	3%	1.6
OMNet++	-	-	2%	-	2%	0.8
Others (user written program)	2%	5%	10%	11%	3%	6.2
Total	100%	100%	100%	100%	100%	

III. SIMULATION METHODOLOGIES AND TECHNIQUES

A. Benchmarking techniques

Benchmarking is a measures of best practice in the presence of fault loads in improving the network performance [24]. The type of fault loads measured, including faults or stressful situations that may caused by incorrect human actions, hardware malfunction or software errors. The benchmarking is used to compute the resiliency, dependency, cost and performance of the networks to present a resolution under a clear set of fault loads. The impact of human failure should be measured as well in the benchmarking process [25]. The three network benchmarking tools are briefly described below.

i) Hpcbench: According to Huang, Bauer and Katchabaw , Hpcbench was developed to measure TCP and UDP performance on high performance networks. Hpcbench can track and record system statistics. The experiment results facilitate comprehensive analysis of network behaviors [26].

ii) NetBench: NetBench consists of nine applications that represent commercial applications for network processors. These applications are from all levels of packet processing; large application level programs as well as small, low-level code fragments are included in the suite [27].

iii) Passmark advanced network test: The Passmark advanced network test is used to test the data transfer rate between two computers in a network. During the test, one of the computers used as client while the other one was server. During the process, the client connects the server and sends continuous data. This network benchmarking test can work with TCP/IP, including asymmetric digital subscriber line (ADSL**)**, Ethernet, cable modems, dial-up modems, wide area networks (WANs), local area networks (LANs), and wireless networks [28].

B. Simulation methodologies

An ideal simulator should model all aspects of the network, is easy to modify, run simulation model faster, and produce credible results. Getting the proper level of abstraction is important since increasing the simulator's accuracy almost always comes directly at the expense of speed. In the remainder of this section, we briefly review previous work on the approaches in improving simulation credibility, methodology and techniques.

Simulator validation and accuracy: Simulation and benchmarking software can be validated using aspects such as general feature, visual, coding, efficiency, modeling assistance, testability, input/output, financial and technical features, user support, and the pedigree [29]. Table III lists the 11 evaluation aspects of network simulators [30].

Select benchmarks and input sets: The SixSigma proposes three steps in completing a benchmarking process. The following are the steps that have been adopted in networks [24]. The first step is to measure and evaluate the operation/process in the networks to identify its strengths and weaknesses. The second step is to initiate a benchmarking study by comparing the processes and results that are more productive with the current networks. The third step is to determine how the successful processes and procedures from the benchmark data can be adapted to the current network processes [24].

TABLE III. SIMULATOR EVALUATION ASPECTS

Simulation aspect	Explanation
General feature	Evaluates the general features of simulators such as the type of simulation (discrete/continuous), ease of use, and user friendliness.
Visual	Evaluates the quality of the graphical representation of the simulation models such as icons and animation.
Coding	Evaluates the flexibility and robustness of the software in allowing additional coding.
Efficiency	Evaluates the capability and effectiveness in modeling variety of complex systems.
Modeling assistance	Evaluates the type and level of assistance provided by the software such as online help.
Testability	Evaluates the facilities for model verification such as error messages, and provision of steps function.
Software compatibility	Evaluates whether the software can be integrated or interfaced with other software such as a benchmarking tool.
Input/Output	Evaluates whether external data can be used with the simulator and also the quality of the output data.
Financial and technical feature	Evaluates the cost and technical features of a simulator such as installation and maintenance issues.
User support	Evaluates the quality of support provided by the supplier such as technical support and updating of products.
Pedigree	Evaluates the origin of the simulator, its distribution and also reputation.

Simulation: The next step after evaluating the simulator and selecting the benchmark is to conduct the simulation process itself [31]. The main criteria of a network simulator is the ability to accurately match the generated network model to the real life network topology [13].

Performance analysis: The final step in the network simulation process is to evaluate and analyze the results [31]. Two guidelines are proposed in [1] to evaluate the results. The first is to ensure that the reported simulation result is repeatable. The second is to specify the appropriate method to analyze the simulation results and errors associated with the results.

C. Simulation techniques

In running a network simulation, there are three widely used techniques: (1) parallel; (2) distributed; and (3) a combination of both parallel and distributed [32].

The parallel and distributed simulator can also be further classified either stochastic or discrete-event simulation. Stochastic simulation is defined as the simulation of random processes which is regarded as a statistical experiment in which the data is analyzed using some statistical methods [1]. Discrete-event simulation on the other hand is a model developed to observe the time based behavior of a system [33].

Network simulation have mostly been performed on small network models, and for short time scales due to performance limitation of the current network simulators [34]. Over the years, the network models have grown in size and complexity therefore increasing the execution time of network simulation [35].

Parallel simulation is the term used to describe the process of synchronizing several simulations that are running on multiple inter-connected processors correctly [36]. To achieve this, parallel executions have to be accurately synchronized to maintain the right dependencies and orderings throughout the evaluation of simulation across processors [36].

There are still many issues concerning distributed and parallel network simulation as new techniques developed. Kiddle, Simmonds and Unger [37] mentioned the issue of designing fast parallel isolated event simulation system for parallel computers with shared memory. The process can be simplified by using common memory space than using only message-passing. The second issue is the development of easy to use network simulator supporting parallel networks. A practical approach to build a parallel network simulator is to base its development on a popular sequential network simulator (NS) as it will minimize the learning time and it can also support large-scale networks [35]. Simulation of large networks requires a huge amount of memory and processing time. One way of speeding up these simulations is to distribute the model over a number of connected workstations. However, this introduces inefficiencies caused by the need for synchronization and message passing between machines. In distributed network simulation, one of the factors affecting message passing overhead is the amount of cross-traffic between machines [38].

One of the many challenges in distributed and parallel network simulation is the minimization of runtime execution overheads such as computation, memory, and communication acquired during the parallel execution [36]. If the distributed and parallel network simulation is run as a discrete event simulation, users have to take into account the extra time to run the whole process. Mota et al. [39] mentioned that discrete-event simulation is often time-consuming process because the telecommunication networks are becoming increasingly complex and large number of observations are essential to get the precise results. To reduce the time duration of simulation, the authors suggested running multiple replications in parallel (MRIP) concurrently on a number of machines [39].

D. Simulation credibility

The most important aspect of network simulation or modeling is its ability to accurately model the real network topology. The simulator should be able to model events such as link change, route change, link failure, and link overloading [13].

The credibility of the simulation software is an important issue when assessing a network. The execution of various processes during simulation run may affect the final results. A simulation process should represent the actual network environment being evaluated. Although most of the articles focused on the importance of simulation software credibility, only a handful of articles have actually addressed the issues of choosing credible simulation tool. For example, Hlupic et al. [30] provided a detailed guidelines for selecting credible simulation software. By following these guidelines one can obtain a credible simulation tool for network analysis and performance modeling [1, 30]. Hlupic et al. mentioned that simulation software should be evaluated based on the aspects listed in Table III.

Bowdon [40] provides an insight into the study of network simulation in the early 1970s. The simulations were used to evaluate the adequacy of throughput, resource utilization and turnaround time when measuring system performance [41]. Bowdon stressed that a simulation model must be accurate both statistically and functionally, to ensure valid assessment of the real networks. Although this is an earlier work, it indeed provides some insight into the current progress in network simulation. The importance of credibility of simulation studies is highlighted in recent years. For example, Pawlikowski et al. [1] reported that about 77% papers on simulation of telecommunication networks published in networking literature (1992-1998) were not concerned about the randomness of simulation results. There was no firm indication whether the final results based on an appropriate statistical analysis or the results reported were purely based on randomness [1]. A similar issue is also highlighted by the mobile ad-hoc network research community [42]. Therefore, the credibility aspects of the simulation of telecommunication networks cannot be neglected [43].

Barcellos et al. [44] suggested that a network simulation study needs to be accompanied by an experimental evaluation to ensure that the data collected are more accurate and more credible. However, this could be a problem for institutions with tighter budget since experimental evaluation requires resources to replicate the actual networking environment.

Another important issue is the credibility of both the open source and commercial network simulators. Nieuwelaar and Hunt [45] stated that most open source network simulation tools focus only on the statistics and measurement of the results.

For assessing the credibility of simulation software, it is suggested that the results generated from a simulator should be repeatable, and the method selected in analyzing the result and the statistical errors occurred during simulation study should also be reported.

IV. Recommendations and Future Research

We first highlight three recommendations that add scientific rigor to the simulation process, and then describe three avenues for future work. The three recommendations are as follows:

i) **Choose a credible simulator for simulation tasks:** It is important for computer and telecommunication network researchers and developers to choose a good simulator which offers flexibility in model construction and validation. A good simulator incorporates appropriate analysis of simulation output data, reliable pseudo-random number generators, and statistical accuracy of the simulation results.

ii) **Build valid and credible simulation models**: A main concern in network simulation or any simulation efforts is to ensure a model is credible and represents reality. If this can't be guaranteed, the model has no real value and can't be used to network simulation and modeling [46]. Therefore, after selecting a good simulator for network simulation tasks, it is also important to have a valid and credible simulation model. The validation process begins during the initial stages of a simulation project and continues throughout. Simulation inputs, both qualitative and quantitative, must be examined and validated. In addition to analyzing model inputs, outputs also need to be validated. This is often believed to be a more crucial form of validation. In situations where a model is developed for an existing system, validity tests become statistical comparisons. Data collected from actual system operation can be used as a benchmark for the model.

iii) **Statistical approaches should be used to help reduce the number of simulation and to analyze the simulation results:** The final result in a simulation study must also be considered within the context since modeling only yields approximate answers. The random number generators used to drive most models provide estimated characteristics. Statistics must be used as a tool for interpreting output.

Future work on simulation methodology should proceed along three avenues. First, to conduct a comprehensive study on the current network simulators and categorizing them based on their performance. This requires a thorough evaluation and experimentation of each of the simulators based on the detail criteria as suggested by Heupic et al. [30]. The findings from this study could be used by organizations and industries in choosing an appropriate simulator for their settings.

Second, research is required on improving the network simulation methodology. This requires an in-depth investigation because the current research on network simulation methodologies is not adequate. Data collected from the industry could be use for analysis on how simulations are being conducted in the actual environment. Third, to conduct study on the business value and practicality of running the network simulation compared to the network experiment.

V. Concluding Remarks

This paper reviewed simulation of telecommunication networks. The network simulators were compared based on type and deployment mode. The simulation evaluation techniques are discussed. A comprehensive survey of 8370 papers published in the selected IEEE Journal and Conference proceedings reveals that majority network researchers are using ns-2 for simulation tasks.

We also emphasized the importance of using a good simulator for network simulation and modeling tasks. Telecommunication network researchers and developers should be aware of the credibility of simulation tools. A credible simulator offers more flexibility in model development, modification and validation, and incorporates appropriate analysis of simulation output data, pseudo-random number generators, and statistical accuracy of the simulation results.

There are several interesting research problems in the area of network simulation. Some of these research issues include, a comprehensive study on the current network simulators and categorizing them based on their performance, improving on network simulation methodologies, a study on the business value and practicality of running the network simulation. We are currently addressing some of these research problems, and research results will be presented in future articles.

In this paper we provide three specific recommendations. Specifically, network researchers should: (1) choose a credible network simulator for simulation tasks; (2) build valid and credible simulation models; and (3) use statistical approaches to improve credibility of simulation results. Adopting these suggestions will help to produce a sound, scientific underpinning for computer network research.

REFERENCES

[1] K. Pawlikowski, H.-D. J. Jeong, and J.-S. R. Lee, "On credibility of simulation studies of telecommunication networks," *IEEE Communications Magazine*, vol. 40, no. 1, pp. 132-139, 2002.

[2] A. M. Law and W. D. Kelton, *Simulation modelling and analysis*, third ed. New York: McGraw-Hill, 2000.

[3] B. Schmeiser, "Simulation output analysis: A tutorial based on one research thread," presented at the 2004 Winter Simulation Conference, December 5-8, 2004, pp. 162-170.

[4] R. Fantacci, T. Pecorella, and I. Habib, "Proposal and performance evaluation of an efficient multiple-access protocol for LEO satellite packet networks," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 3, pp. 538-545, 2004.

[5] G. Yuehong, Z. Xin, Y. Dacheng, and J. Yuming, "Unified simulation evaluation for mobile broadband technologies - [Modeling and simulation: a practical guide for network designers and developers]," *IEEE Communications Magazine*, vol. 47, no. 3, pp. 142-149, 2009.

[6] M. Niazi and A. Hussain, "Agent-based tools for modeling and simulation of self-organization in peer-to-peer, ad hoc, and other complex networks," *IEEE Communications Magazine*, vol. 47, no. 3, pp. 166-173, 2009.

[7] J. S. Carson II, "Introduction to Modeling and Simulation," presented at 2004 Winter Simulation Conference, December, 2004, pp. 1283-1289.

[8] K. Pawlikowski, "Steady-state simulation of queuing processes: a survey of problems and solutions," *ACM Computing Surveys*, vol. 1, no. 2, pp. 123-170, 1990.

[9] *OPNET Technologies*. Retrieved February 10, 2011, from www.opnet.com

[10] X. Chang, "Network simulation with opnet," presented at 1999 winter simulation conference, Phoenix, AZ, USA, 1999, pp. 307-314.

[11] Scalable Network Technologies. *QualNet Developer*. Retrieved March 5, 2011, from http://www.qualnet.com/products/developer.php

[12] Tetcos. *Products*. Retrieved March 5, 2011, from http://www.tetcos.com/software.html

[13] J. Fritz. *Network-modelling tools*. Retrieved March 5, 2011, from http://www.techworld.com/networking/features/index.cfm

[14] Shunra. *Shunra Virtual Enterprise (Shunra VE) 5.0*. Retrieved January 10, 2011, from http://www.shunra.com/content.aspx?pageId=69

[15] K. Fall and K. Varadhan. *The ns manual. The VINT project*. Retrieved February 10, 2011, from http://www.isi.edu/nsnam/ns/doc/

[16] *CMU Monarch Project*. Retrieved January 5, 2011, from http://www.monarch.cs.cmu.edu

[17] M. Greis. *Ns-2 tutorial*. Retrieved January 5, 2011, from http://www.isi.edu/nsnam/ns/tutorial/index.html

[18] GloMoSim. *GloMoSim Manual*. Retrieved February 10, 2011, from http://pcl.cs.ucla.edu/projects/glomosim/GloMoSimManual.html
[19] *OMNeT++*. Retrieved March 5, 2011, from http://www.omnetpp.org/
[20] N. Kotilainen, M. Vapa, T. Keltanen, A. Auvinen, and J. Vuori, "P2P Realm - Peer-to-peer network simulator," presented at the 11th International Workshop on Computer-Aided Modeling, Analysis and Design of Communication Links and Networks, June 8-9, 2006, pp. 93-99.
[21] MANIACS. *GTNETS*. Retrieved March 5, 2011, from http://www.ece.gatech.edu/research/labs/MANIACS/GTNetS/
[22] G. Ewing, K. Pawlikowski, and D. McNickle, "Akaroa 2: exploiting network computing by distributed stochastic simulation," presented at European Simulation Multiconference (ESM'99), Warsaw, Poland, June, 1999, pp. 175-181.
[23] *OMNEST*. Retrieved March 5, 2011, from www.omnest.com
[24] SixSigma. *Is benchmarking and identifying best practices worth It?* Retrieved March 5, 2011, from http://www.isixsigma.com/library/content/
[25] C. Constantinescu, K. Kanoun, H. Madeira, B. Murphy, I. Pramanick, and A. B. Brown, "Panel Statement," presented at the International Conference on Dependable Systems and Networks, 2005, pp.
[26] B. Huang, M. Bauer, and M. Katchabaw, "Hpcbench - A linux-based network benchmark for high performance networks," presented at Proceedings of the 19th international symposium on high performance computing systems and applications, 2005, pp. 65-71.
[27] G. Memik, W. H. Mangione-Smith, and W. Hu, "Netbench: A benchmarking suite for network processors," presented at IEEE/ACM International Conference on Computer Aided Design, 2001, pp. 39-42.
[28] Passmark. *Network benchmark - test your network speed*. Retrieved March 5, 2011, from http://www.passmark.com/products/pt_advnet.htm
[29] X. Liu and A. A. Chien, "Realistic Large-Scale Online Network Simulation " presented at Proceedings of the 2004 ACM/IEEE conference on Supercomputing 2004, pp. 31.
[30] V. Hlupic, Z. Irani, and R. J. Paul, "Evaluation framework for simulation software," *International journal of advanced manufacturing technology*, vol. 15, no. 366-382, 1999.
[31] J. J. Yi and D. J. Lilja, "Simulation of computer architectures: simulators, benchmarks, methodologies and recommendations," *IEEE Transaction on Computers*, vol. 55, no. 3, pp. 268-280, 2006.
[32] L. Breslau, D. Estrin, K. Fall, S. Floyd, J. Heimann, A. Helmy, P. Huang, S. McCanne, K. Varadhan, Y. Xu, and H. Yu, "Advances in network simulation," *Computer*, vol. 33, no. 5, pp. 59-67, 2000.
[33] P. Ball. *Introduction to discrete event simulation*. Retrieved March 5, 2011, from http://www.dmem.strath.ac.uk/~pball/simulation/simulate.html
[34] S. Bhatt, R. Fujimoto, A. Ogielski, and K. Perumalla, "Parallel simulation techniques for large-scale networks," in *IEEE communication magazine*, 1998, pp. 42-46.
[35] Y. Li and D. Qian, "A practical approach for constructing a parallel network simulator," presented at Proceedings of the Fourth International Conference on Parallel and Distributed Computing, Applications and Technologies, 27-29 Aug, 2003, pp. 655-659.
[36] K. Perumalla, "Parallel and distributed simulation: traditional techniques and recent advances " presented at Proceedings of the 37th conference on Winter simulation Monterey, California 2006, pp. 84-95.
[37] C. Kiddle, R. Simmonds, and B. Unger, "Performance of a mixed shared/distributed memory parallel network simulator," presented at the eighteenth workshop on Parallel and distributed simulation, Kufstein, Austria, 2004, pp. 17-25.
[38] S. Lee, J. Leaney, T. O'Neill, and M. Hunter, "Performance benchmark of a parallel and distributed network simulator," presented at the 19th Workshop on Principles of Advanced and Distributed Simulation, 2005, pp. 101-108.
[39] E. Mota, A. Wolisz, and K. Pawlikowski, "A perspective of batching methods in a simulation environment of multiple replications in parallel " presented at the 32nd Conference on Simulation Orlando, Florida 2000, pp. 761-766.
[40] E. K. Bowdon, "Using simulation to evaluate system performance," presented at Proceedings of the 11th workshop on Design automation, 1974, pp. 359-365.
[41] E. K. Bowdon, "Using simulation to evaluate system performance " presented at Proceedings of the 11th workshop on Design automation 1974, pp. 359-365.
[42] S. Kurkowski, T. Camp, and M. Colagrosso, "MANET simulation studies: The incredibles," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 9, no. 4, pp. 50-61, 2005.
[43] M. C. Weigle, "Improving confidence in network simulations," presented at the 37th conference on Winter simulation, Monterey, California, 2006, pp.
[44] M. P. Barcellos, G. Facchini, and H. H. Muhammad, "Bridging the gap between simulation and experimental evaluation in computer networks," presented at the 39th Annual Symposium on Simulation, 2006, pp. 286-293.
[45] M. van den Nieuwelaar and R. Hunt, "Real-time carrier network traffic measurement, visualisation and topology modelling," *Computer Communications*, vol. 27, no. 1, pp. 128-140, 2004.
[46] R. McHaney, *Computer simulation: a practical perspective*. San Diego: Academic Press, 1991.

Nurul I Sarkar is currently a Senior Lecturer in the School of Computing and Mathematical Sciences at AUT University, New Zealand. He has a PhD in Electrical and Electronic Engineering from The University of Auckland and his current research interests include wireless networks, cross-layer design optimization, and network security. Dr Sarkar currently serves on editorial boards on several international journals. He is a Senior Member of IEEE. Dr. Sarkar is the author of more than 80 refereed journal and conference papers. He served as TPC co-chair for mobile and wireless networking track (IEEE ATNAC 2010 and TENCON). He served as TPC member for various leading networking conferences as well as track and session chairs for several national and international forums. He also serves as Guest Editor for AP Journal of Networks.

S. A. Halim is with the Computer Sciences Department, Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA Perlis, Malaysia.