Australian Journal of Basic and Applied Sciences, 6(1): 143-148, 2012
ISSN 1991-8178

Performance Evaluation in MANETs Environment

Mohammad M. Qabajeh, Aisha-Hassan A. Hashim, Othman O. Khalifa, Liana K. Qabajeh and Jamal I. Daoud

Faculty of Engineering, Department of Electrical and Computer Engineering, International Islamic University Malaysia (IIUM).

Abstract: Mobile Ad hoc NETwork (MANETs) are collections of mobile nodes that communicate with each other over wireless links in the absence of any infrastructure or centralized administration. The main characteristics of MANETs are limited bandwidth, limited resources, dynamic mobility, and high contention radio medium. In MANETs, the mobile nodes may be laptops, palmtops, PDAs, mobile phones, or pocket PC with wireless connectivity. Mobility and high cost of these devices makes performance evaluation of this type of environment not easy task. This paper presents the different performance evaluation methods used by the research community. Simulation evaluation method will be discussed in detail since it is considered as one of the most popular evaluation methods. Along with this, a valuable description of the current simulation tools will be provided. In short, this paper provides a road map for researchers in choosing the proper evaluation method and choosing the most a appropriate simulation tool.

Key words: MANETs, performance evaluation, simulation tools.

INTRODUCTION

In the early 1972, DARPA starts its research in Ad hoc networks by deploying its Packet Radio Networks (PRnet) (Kahn, Gronemeyer, Burchfiel, & Kunzelman, 1978). Since that time, the concept of Ad hoc wireless networks is introduced. Ad hoc networks are formed when a collection of mobile devices communicate with each other without pre-established infrastructure. Nodes in Ad hoc network are often mobile, but it can also consist of stationary nodes (Mahmoud, 2005). Each of the nodes has a wireless interface and communicates with others over either radio or infrared channels.

Mobile Ad hoc NETwork (MANET) is a type of Ad hoc networks with rapidly changing topology. Formally, MANETs are collections of mobile nodes that communicate with each other over wireless links in the absence of any infrastructure or centralized administration (Chlamtac, Conti, & Liu, 2003). This ensures that the network will not cease functioning just because one of the mobile nodes moves out of the range of the others. Each mobile node acts as a host generating flow, being the receiver of a flows from other mobile nodes, or as a router and responsible for forwarding flows to other mobile nodes (Lin T. , 2004) (Mohapatra & Krishnamurthy, 2005). Mobile nodes in Ad hoc networks have a limited transmission range, nodes that relies within the transmission range can communicate directly with each other, while intermediate nodes is needed to forward flow between nodes that are unable to communicate directly.

MANETs are useful in many application environments, where instant deployment and dynamic reconfiguration are necessary and wired infrastructure is not available. Examples of these applications include disaster recovery efforts, military battlefields, conferences and classrooms without the support of a wired infrastructure, and communication among a group of islands or ships. However, this type of networks characterized by limited bandwidth, limited resources, dynamic mobility, and high contention radio medium. These characteristics pose extra challenges to the design of routing protocols.

This paper presents the evaluation methods used in MANETs environment. Then, the details about the different simulation tools are provided as well as a fair comparison among these tools. The remainder of this paper is structured as follows: section 2 gives an overview on the performance evaluation methods. Section 3 presents an overview of some of the simulation tools. Then, a discussion will be provided and finally a conclusion will be proposed.

Performance Evaluation Options:

In MANETs, evaluating and testing the routing protocols is a mandatory phase to its success in a real world application. To perform this evaluation, researchers have four options, using analytical modeling, using simulation tools, using emulators, or using test-beds. Overview about each of these choices is presented below

Testbeds:

Testbeds are often known as in-lab networks built and used by research community. In MANETs, the best way to predict the network behavior is to deploy it in a real environment which provides the best realism.

Corresponding Author: Mohammad M. Qabajeh, Faculty of Engineering, Department of Electrical and Computer Engineering International Islamic University Malaysia (IIUM).
E-mail: m_qabajeh@yahoo.com

Executing the real code on the real environment can detect more details that might be missed in the simulation (Hogie & Bouvry, 2006).

However, testbeds have several drawbacks. First, the cost is very high. Since building a testbed requires mobile terminals, wireless transmits equipment, analysis tools and high intensive labor to monitor the testbeds, which is very expensive. Second, high complexity. Since managing the deployment process, monitoring and wide range of mobility scenarios makes conducting testbeds a challenging task. Third, does not support scalability. The challenges facing constructing testbeds make it difficult to support scalable networks. Fourth, assurance of repeatability, since the nature of wireless environment makes it hard to ensure execution under similar conditions for each test run (Morten & Anders, 2007). It is proven in the literature that no testbeds of more than 50 nodes were proposed (Hogie & Bouvry, 2006) (Newport, 2004) (M. Shahidul, Hongnian, Alison, & T. C., July 6-11, 2008).

Emulation:

An emulator provides a combination of pure simulations and protocol implementation (testbeds), some of the network components are implemented in the real world and the others are simulated. In particular, the purposes of Emulators are to allow testing the protocol in real hardware and to prepare for direct execution of the protocol in the real word. This allows setting and testing some underlying parameters or functionality of the proposed protocol in-lab without physically moving the nodes to the real environment (Jorge, Juan-Carlos, Carlos, & Pietro, 2009). For example, the properties of a physical layer can be emulated using Field Programmable Gate Arrays (FPGAs) (Judd & Steenkiste, 2004).

Emulators have several advantages. First, using hybrid between network simulation and protocol implementation provide accurate evaluation of the MANET protocol in low cost and large scale. Second, emulation is closer to realism compared with simulation, since mobile nodes and traffic pattern is real while only the link pattern is emulated (David A.Maltz, 1999). Third, the cost of emulation is low, since it can be built in the lab environment and no additional equipment is needed. Forth, emulation provides seamless connection between protocol evaluation and implementation. Since the software used in emulation can directly used in the testbed.

Analytical Modeling:

Analytical models use mathematical notions and models to describe certain aspects of a system. Generally, it provides a best qualitative insight into the effects of various parameters and their interactions. Analytical modeling is a cost effective evaluation method because it gives a thorough understanding of the system. Also, it can often be quickly set up and evaluated, requires only paper, pencils and time to analyze the model. Therefore, analytical modeling is the cheapest performance evaluation method.

However, analytical models cannot capture all the details that can be built into simulation models. Simulations can incorporate more details and require less assumptions than analytical modeling and, thus, more often are closer to reality. In addition, a solid mathematical background and probability theory are needed to build this kind of models. Many systems are too complex for analytical modeling, which require simplifications, assumptions and approximations to turn out accepted results (Raj Jain, May 1991).

Simulation:

Simulation is generally defined as the process of designing a model of a real system and conducting experiments with this model for the purpose of understanding the behavior of the system and/or evaluating various strategies for its control (Banks, 1998). Simulation is widely used method in evaluating MANET routing protocols (Stuart, Tracy, & Michael, 2005). This is due to the following reasons. First, because it provides cheap, repeatable and controlled way with acceptable overhead required to carry out a simulation. Second, simulation allows evaluating scalable networks. Third, simulation enables experimentation with configurations that may not be possible with existing technology. Fourth, simulation allows continuous development of the models and can be considered as an early stage of the actual implementation which simplifies the real implementation of the models (Newport, 2004).

However, simulation results are generally not be as accurate as real implementations because implementation can provide more reality than simulation tools. The following section will discuss more details about the simulation tools.

Overview of Simulation Tools:

Currently, several simulation tools exists for Ad hoc networks, including DIANEmu (Klein), REAL (Keshav, December, 1988), GloMoSim (Zeng, Bagrodia, & Gerla, 2002), GTNets (Riley, 2003), Jane (Frey, Gorgen, Lehnert, & Sturm, 27-28 November, 2003), NS-2 (McCanne & Floyd, 1997), pdns (Riley, Fujimoto, and Ammar, 1999), OPNET (Desbrandes, Bertolott, & Dunand, 1993) (OPNET Technologies, 2011), OMNeT++ (OMNeT++, 2011) (OMNeT++, 2011) and SWANS (Barr, 2004). These tools differ in their simulation capabilities, features, environments, scalability …etc. Some are dedicated to MANETs simulation such as (Frey, Gorgen, Lehnert, & Sturm, 27-28 November, 2003) and (Barr, 2004), and some others consist in extensions of wired network simulators such as (McCanne & Floyd, 1997) and general-purpose discrete-event simulation engines such as (Bagrodia, et al., 1998) and (Bagrodia and Liao, 1994).

In the following subsections, we will review five of the most popular simulation tools (Hogie & Bouvry, 2006); Network Simulator 2 (NS-2) (McCanne & Floyd, 1997), Global Mobile Information Systems Simulation (GloMoSim) (Zeng, Bagrodia, & Gerla, 2002), OMNeT++ (OMNeT++, 2011), QualNet (Qualnet, 2011), and Optimized Network Engineering Tools (OPNET) (OPNET Technologies, 2011).

Network Simulator 2 (NS-2):

NS-2 (McCanne & Floyd, 1997) is developed at Information Sciences Institute (ISI), and is supported by the Defense Advanced Research Projects Agency (DARPA) and National Science Foundation (NSF). NS-2 is a discrete-event network simulator organized according to the OSI model and initially intended to simulate wired networks (Hogie & Bouvry, 2006). After that 802.11 MAC Layer and important routing protocols needed in MANETs have been added to it (Schilling, 2005).

The core of NS-2 is a huge piece code written with C++ due to its quickness and Object- oriented possibilities. To ease the use of NS-2, it appears to the user as an OTCL interpreter. It reads scenarios files written in OTCL and produces a trace file in its own format. This trace needs to be processed by user scripts or converted and rendered using the network animator, NAM (Estrin, Handley, Heidemann, McCanne, Xu, & Yu, 1999), which permits to visualize the output, provides packet-level animation, and provides a Graphic User Interface (GUI) interface to design and debug network protocols. The combination of the two languages offers an interesting compromise between performance and ease of use; however this increases the complexity of the simulator and results in a steep learning curve for NS-2 and difficulty in debugging (Cavin, Sasson, & Schiper, 2002).

NS-2 is an open-source simulator, which makes it interesting on the one hand, but on the other hand there are some negative aspects that come along with it. Unfortunately NS-2 suffers from its lack of modularity and its inherent complexity. Indeed, adding components/protocols or modifying existing ones is not as straightforward as it should be.

Learning NS-2 needs a long period of time due to the lack of documentation in the source code and the usage of two programming languages. For a long time, NS-2 has been said to have few good documentation. The situation recently changed, as several users have put online their experience in the form of tutorials or example-driven documentations. Another well-known weakness of NS-2 is its high consumption of computational resources. A harmful consequence is that NS-2 lacks scalability, which impedes the simulation of large networks; NS-2 is typically used for simulations consisting of no more than a few hundreds of nodes (Hogie & Bouvry, 2006) (Schilling, 2005).

Global Mobile Information System Simulator (GloMoSim):

GloMoSim is a scalable simulation environment for wireless and wired network systems that was developed at University of California, Los Angeles (UCLA). GloMoSim is aimed at stimulating models that may contain as many as 100,000 mobile nodes with a reasonable execution time (Bajaj, Takai, Ahuja, Tang, Bagrodia, & Gerla, 1999). It is the second most popular wireless network simulator. GloMoSim is written in the parallel discrete-event simulation capability provided by a C-based parallel simulation language; Parallel Simulation Environment for Complex systems (PARSEC) (Bajaj, Takai, Ahuja, Tang, Bagrodia, & Gerla, 1999) and hence benefits from the latter's ability to run on shared-memory symmetric processor computers.

GloMoSim respects the Open Systems Interconnection model (OSI) standard and has been developed using languages, libraries and frameworks dedicated to discrete-event simulation. These middleware technologies typically focus on performance, concurrency and distribution (Hogie & Bouvry, 2006). Standard Application Programming Interfaces (APIs) are used between the different layers. This allows the rapid integration of models developed at different layers by different users (Cavin, Sasson, & Schiper, 2002). Two versions of the simulation tool exist: the academic research version, which is for academic use only, and a commercial version, which is distributed as the QualNet software package.

GloMoSim uses parallelism which refers to the simultaneous execution of different instructions of the same program. Parallelism is used to quicken simulations and allow GloMoSim to model networks made of tens of thousands stations (Hogie & Bouvry, 2006). The parallelization technique used by GloMoSim is to split the network into different sub-networks, each of them being simulated by distinct processors. The network is partitioned in such a way that the number of nodes simulated by each partition is homogeneous.

Source Code is written primarily in C and the PARSEC compiler is used to create executable files. For the development of custom protocols in GloMoSim, some familiarity with PARSEC is required. Most protocol developers will write purely C code with some PARSEC functions for time management. PARSEC code is used extensively in the GloMoSim kernel, but it is not required to know and understand how the kernel works (Christiansen, Kuijpers, Yomo, & Fathi, 2003).

Qual Net:

Scalable Network Technologies developed QualNet (Qualnet, 2011) as a commercial version of GloMoSim network simulator which is mainly used for wireless networks. QualNet offer more feature than GloMoSim, these features include extensive documentation and technical support, user-friendly tools, tools for building scenarios and analyzing simulation output and offers large set of modules and protocols for both wired and wireless networks (local, Ad hoc, satellite and cellular). QualNet runs on all common platforms (Linux, Windows, Solaris). Since QualNet built on top of GloMoSim, it is written in PARSEC (Bajaj, Takai, Ahuja, Tang, Bagrodia, & Gerla, 1999). PARSEC is used to provide event scheduling and parallel simulation services.

There are three libraries available of QualNet. A *standard library* which offers most of the protocols and models required for research and business-oriented activities in both wired and wireless networks. A *MANET library* which provides very specific additional components for Ad hoc networks other than those already present in the standard library. A *QoS library* which includes quality-of-service specialized protocols. Also, QualNet includes a Digital Elevation Model (DEM*)* to make nodes and radio waves moving in non-flat terrains with specified radio absorption characteristics. QualNet seems to be the most complete network simulator, in terms of available protocols, models and tools for what concerns mobile Ad hoc networks (Caro, December 2003).

Optimized Network Engineering Tools (OPNET):

OPNET (Desbrandes, Bertolott, & Dunand, 1993) (OPNET Technologies, 2011) is a discrete-event network simulator first proposed by Massachusetts Institute of Technology (MIT) in 1986. OPNET is written in C++ and is a well-established and professional commercial suite for network simulation. It is actually one of the most widely used commercial simulation environment (Hogie & Bouvry, 2006). One of the most interesting features of OPNET is its ability to execute and monitor several scenarios in a concurrent manner.

OPNET comes along with a large number of predefined functions, protocols, devices and behaviors, which make it a powerful program just from the start up and without big effort. Additionally, the opportunity to implement new algorithms is given. Also, several tools and editors are provided. The aim is to make use of the numerous existing components that are part of OPNET in order to decrease the developers' effort, the shrink implementation time, and reduce the number of errors. OPNET provides a hierarchal GUI feature and a lot of documentation comes along with it.

Nevertheless, OPNET is not open-source software and therefore users and companies need to purchase licenses. Hence, the cost of the software could discourage many developers, since open-source solutions are available (Schilling, 2005). Additionally, the main disadvantage is its relative complexity to model a given system. The time required to learn it and achieve the modeling of a system can be very long, especially for new developments (Christiansen, Kuijpers, Yomo, & Fathi, 2003). Furthermore, it is reported that the OPNET simulator is pretty memory consuming and that it is difficult to modify the library models.

OMNeT++:

OMNeT is a discrete event simulator has been publicly available since 1997. In particular, it is a general-purpose simulator capable of simulating any system composed of devices interacting with each others. OMNeT use C++ programming language and use object-oriented design.

OMNeT have been used in several research areas including wireless and ad-hoc networks, sensor networks, IP and IPv6 networks, multiprocessors and other distributed hardware systems, wireless channels, peer-to-peer networks, storage area networks (SANs), optical networks, queuing networks, file systems, validating hardware architectures, high-speed interconnections (InfiniBand), and others (Imre, Keszei, Hollós, Barta, & Kujbus, 2001). In general, OMNeT is not designed especially for telecommunication networks.

OMNeT is a component based simulator and the basic entity in OMNeT++ is a module. Modules are composed of submodules or they can be atomic. The atomic modules capture the actual behavior. Modules communicate with each other via messages through gates. Gates are linked to each other using connections. For example, the protocol models can be combined into a compound module representing a host node (Imre, Keszei, Hollós, Barta, & Kujbus, 2001).

Discussion:

MANETs simulators have different features and models, so selecting the proper simulation tool depends on several factors. First, choosing a simulation package depends on the research requirement. The availability of the routing protocols to be simulated and support of the under investigation problem are of great importance. Moreover, the number of nodes targeted also determines the choice of the simulation tool. Sequential simulators should not be expected to run more than 1,000 nodes. If larger scales are needed, then parallel simulators are a wise choice. Table 5.1 summarizes the properties of the five discussed simulation tools.

Since OPNET is a commercial tool, purchase of the software and the model libraries are expensive. In addition, it suffers from many disadvantages such as complexity, time required to learn it, memory consumption, and difficulty to modify the library models. The most elaborate tools are GloMoSim and NS-2. Both, the latter and the academic research version of the former are available freely on the Internet. In summary, the following comments are concluded:

- NS-2 is the most famous simulator. However, it is quite a complex task to install it and have it works right. Even after installing it, it is difficult to learn and use specially that it uses two languages C++ for data and OTcl for control. There is no clear separation between C++ and OTcl.
- GloMoSim on the other hand is built as a set of libraries. The libraries are built in a C-based discrete event simulation language (PARSEC). Even the new protocols are developed using C language; C language is more familiar than OTcl for most programmers.
- NS-2 does not work well for large topologies; it can be used for simulation of the order of hundreds of nodes. GloMoSim however is a scalable simulator that is able to model networks made of tens of thousands nodes.
- GloMoSim is considered as the second famous simulator after NS-2 simulator (Hogie & Bouvry, 2006).

Table 1: Comparison between Ad hoc simulators.

Feature \ Tool	NS2	GloMoSim	QualNet	OPNET	OMNeT++
Interface	C++/OTCL	Parsec(C-based)	Parsec (C-based)	C	C++
Parallelism	No	Yes	Yes	Yes	Yes
Popularity	high	moderate	low	low	low
License	Open source, some extensions require license	Open source	Commercial (relatively expensive)	Free for academic and educational use/ Commercial(relatively expensive)	Free for academic/ educational use/ Commercial
Documents and user support	Excellent	Poor	Excellent	Excellent	Good
Required time to learn	long	Moderate	Easy to learn	long	Moderate
Scalability	moderate	high	high	high	moderate
Extendibility	Excellent	Excellent	Excellent	Excellent	Excellent
Graphic interface support	Limited GUI	Limited GUI	Excellent GUI	Excellent GUI	Good GUI

Conclusion:

In this paper we have discussed the performance evaluation methods used in the research community of wireless Ad hoc networks environment. This paper also presents detailed information about some of the most popular network simulators and we highlight their strengths and weaknesses. The recommendations presented in this paper helps the researchers in Ad hoc networks field in selecting the most appropriate evaluation method as well as selecting the simulation tool.

REFERENCES

Bagrodia, R. and W. Liao, Maisie: A language for the design of efficient discrete-event simulations. IEEE Tranactions in Software Engineering, 20(4): 225-238.

Bagrodia, R., R. Meyer, T.M, C.Y, X. Zeng, J. Martin, *et al.* 1998. Parsec: A parallel simulation environment for complex systems. IEEE Computer, 31(10): 77-85.

Bajaj, J., M. Takai, R. Ahuja, K. Tang, R. Bagrodia and M. Gerla, 1999. *Glomosim: A scalable network simulation environment.* Technical Report, Computer Science Department, University of California, Los Angeles.

Banks, J., 1998. *Handbook of Simulation : Principles, Methodology, Advances, Applications, and Practice.* USA: John Wiley and Sons, Inc.

Barr, R., 2004. *An efficient, unifying approach to simulation using virtual machines.* PhD thesis, Faculty of the Graduate School, Cornell University.

Caro, G.A., 2003. *Analysis of simulation environments for mobile ad hoc networks.* Technical Report , Dalle Molle Institute for Artificial Intelligence, Switzerland.

Cavin, D., Y. Sasson, A. Schiper, 2002. On the accuracy of MANET simulators. *Proceedings of the second ACM international workshop on Principles of mobile computing*, Toulouse, France., 38-43.

Chlamtac, I., M. Conti, and J. Liu, 2003. Mobile ad hoc networking: imperatives and challenges. *Elsevier, Ad Hoc Networks , 1*, 13-64.

Christiansen, H., G. Kuijpers, H. Yomo and H. Fathi, 2003. *Simulator requirements, comparative evaluation of tools.* Technical University of Denmark, CNTK.

David, A., J.B. Maltz, 1999. *Experiments Designing and Building a Multi-Hop Wireless Ad Hoc Network Testbed.* Technical Report, School of Computer, Carnegie Mellon University.

Desbrandes, D., S. Bertolott, and L. Dunand, 1993. Opnet 2.4: An environment for communication network modeling and simulation. *In Proceedings of European Simulation Symposium. Society for Computer Simulation,* pp: 64-74.

Estrin, D., M. Handley, J. Heidemann, H. McCanne, X. Xu, and H. Yu, 1999. *Network visualization with the VINT Network Animator Nam.* Technical Report, USC Computer Science Department.

Frey, H., D. Gorgen, J. Lehnert, and P. Sturm, 27-28 November, 2003. A java-based uniform workbench for simulating and executing distributed mobile applications. *In Proc. of the FIDJI International Workshop on Scientific Engineering of Distributed Java Applications*, pp: 1-12). Luxembourg.

Hogie, L. and P. Bouvry, 2006. An Overview of MANETs Simulation', Electronic Notes in Theoretical Computer Science. Elsevier, 150: 81-101.

Imre, S., C. Keszei, D. Hollós, P. Barta, and C. Kujbus, 2001. Simulation environment for ad-hoc networks in omnet++. *IST Mobile Summit*, pp: 135-40.

Jorge, H., C. Juan-Carlos, T.C. Carlos, and M. Pietro, 2009. Testing Applications inMANET Environments through Emulation. *EURASIP Journal onWireless Communications and Networking,* ID 406974: 1-20.

Judd, G., and P. Steenkiste, 2004. Repeatable and realistic wireless experimentation through physical emulation. *ACM SIGCOMM Computer Communication Review* , 34(1): 63-68.

Kahn, R., S. Gronemeyer, J. Burchfiel and R. Kunzelman, 1978. Advances in Packet Radio Technology. *Proceedings of the IEEE* , 66(11): 1468-1496 .

Keshav, S., 1988. *REAL: a network simulator, University of California,.* Technical Report, University of California, Berkeley, Calif, USA.

Klein, M., *Dianemu: A java based generic simulation environment for distributed protocols.* Technical Report, University of Karlsruhe, Faculty of Informatics, Germany.

Lin, T., 2004. *Mobile Ad-hoc Network Routing Protocols: Methodologies and Applications* . Ph.D. thesis, Faculty of the Virginia Polytechnic Institute and State University, Blacksburg, Virginia.

Shahidul, M.H., Y. Hongnian, G. Alison, and T.C, Y, 2008. Co-simulation framework for Networked Control Systems over multi-hop mobile ad-hoc networks. *The International Federation of Automatic Control,* pp: 12552-12557. Seoul, Korea.

Mahmoud, F., 2005. *Reputed Authenticated Routing for Ad Hoc Networks Protocol (Reputed-ARAN).* Mater Thesis, The American University, Cairo.

McCanne, S., and S. Floyd, 1997. *The UCB/LBNL/VINT network simulator.* Available at:http://www.isi.edu/nsnam/ns/, Lawrence Berkeley Laboratory.

Mohapatra, P. and S. Krishnamurthy, 2005. *AD HOC NETWORKS: Technologies and Protocols.* Springer.

Morten, J. and N. Anders, 2007. *Evaluation of Routing Dependability in MANETs using a Topology Emulator.* Master thesis, Department of Control Engineering at the Institute of Electronic Systems at Aalborg University, Denmark.

Newport, C., 2004. *Simulating mobile ad hoc networks:A quantitative evaluation of common manet simulation models.* Technical Report, In Dartmouth College Computer Science .

OMNeT++, 2011. Retrieved from OMNeT++ Network Simulation Framework: http://www.omnetpp.org

OPNET Technologies, 2011. Retrieved from Application and Network performance with OPNET: http://www.opnet.com

Qualnet., 2011. Retrieved from http://www.scalable-networks.com/products/qualnet/

Raj, Jain., 1991. *The Art of Computer Systems Performance Analysis Techniques for Experimental Design, Measurement, Simulation and Modeling.* Wiley-Interscience.

Riley, G., 2003. The georgia tech network simulator. *In Proceedings of the ACM SIGCOMM workshop on Models, methods and tools for reproducible network research (MoMeTools '03), ACM Press,* pp: 5-12.

Riley, G., R. Fujimoto, and A. M., 1999. A generic framework for parallelization of network simulations. In Proceedings of the 7th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS '99), pp: 128-135.

Schilling, B., 2005. *Qualitative Comparison of Network Simulation Tools', Modeling and Simulation of Computer Systems seminar.* Institute of Parallel and Distributed Systems (IPVS), University of Stuttgart.

Stuart, K., C. Tracy and C. Michael, 2005. MANET simulation studies: the incredibles. ACM SIGMOBILE Mobile Computing and Communications Review - Special Issue on Medium Access and Call Admission Control Algorithms for Next Generation Wireless Networks, 9(4): 50-61.

Zeng, X., Bagrodia and R. Gerla, Glomosim: A library for parallel simulation of large-scale wireless networks. *Workshop on Parallel and Distributed Simulation,* pp: 154-161.