

# Key Features and Optimum Performance of Network Simulators: A Brief Study

Atif Mahmood<sup>1</sup>, M. Faisal Saleem<sup>1</sup>, Dr. Ahsan Latif<sup>1</sup>, Muddasar Ahmad<sup>2</sup>

<sup>1</sup>Department of Computer Science, University of Agriculture, Faisalabad, Pakistan.

<sup>2</sup>Department of Computing, Universiti Teknologi Malaysia, Johar Bahru, Malaysia.

**Abstract**— In the network area, to establish a network in real time for any experiment is very difficult. To test a simple problem takes a large amount of time and cost. So it is not quite possible to implement the whole network scenario in a real time. There are many tools which helps network developer to test their network scenarios virtually whether the network is able to work in real time environment. The network simulators are used for performance analysis in the field of communication. With the help of these simulation tools both the time and cost of testing the functionality of network have been reduced and implementations are made easy. In this paper, we briefly discuss the main features of different simulators, new developments and also discuss their advantages and disadvantages in the field of network simulation. We hope that the people who face difficulty in selecting a proper suitable network simulation tool in their research this paper is helpful for them.

**Keywords**— Network Simulation, Network Simulator, OPNET, NS2, NS3, OMNet++.

## I. INTRODUCTION

A network simulation is a virtual representation of a network. There are two common types of network simulation—one is used on existing networks, and the other is used before a network is constructed. The purposes of these simulations are finding problems in an existing network or locating unexpected interactions on one that hasn't been built yet. By locating existing problems or preventing them in the first place, companies can improve network reliability and lower maintenance costs.

In current era, the simulation is one of the most important technologies which simulate the imaginary and real life objects on computer. The basic idea to study the simulator in computer is same as it can model hypothetical and real life objects. The network is also simulated on the computer. To simulate or implement a network on computer a technique is used called network simulation. The basic purpose of this network simulation is to monitor the behavior of the network. The behavior of the network is calculated either by network entities interconnection using mathematical formulas or by capturing and playing back observations from a production network.

*“The Network Simulator provides an integrated, versatile, easy-to-use GUI-based network designer tool to design and*

*simulate a network with SNMP, TLI, TFTP, FTP, Telnet and Cisco IOS device.”[1]*

The researcher used network simulations to test different scenarios that are not easy to present in real life cause of its difficulty and experiments. The network simulators are very useful, providing controlled and reproducible environment to test new networking protocols or to change the existing ones. These simulation tools are also very useful design different network topologies using various types of nodes (routers, bridge, hubs, hosts or mobile units).

The people of different areas such as network designers, industry developers, academic researchers and quality assurances are used to design, verify, simulate and analysing the performance of different network protocols. Generally network simulator tools will include a wide range of networking technologies and protocols which helps user to build complex network from basic building blocks. One can easily design different network topologies using various types of basic building blocks such as end host, mobile users, hubs, network bridges, routers and optical layer designs. These tools are also used to check and evaluate the effect of different parameters on the protocols being observed.

There are different types of network simulators that categorized on the basis of range: (from very simple to very complex). These network simulator are used to

- a) Specifying nodes
- b) Specifying the link between those nodes
- c) Specifying the traffic between these nodes
- d) Specifying everything about the protocols that used to handle traffic in the network.
- e) Visualize the working of network graphical applications
- f) Customization text based applications.
- g) Providing a programming environment that customized to create an application that test the networking environment (programming oriented tools). [2]

Network simulator tools are available with different features. Some of the network simulator are OPNET, NS2,

NS3, NetSim, OMNeT++, REAL, J-Sim and QualNet. In this paper we are working on some of the simulator.

**OPNET (Optimized Network Engineering Tools):** It is extensive and powerful simulation software with wide variety of possibilities to simulate entire heterogeneous networks with various protocols.

**NS2 (Network Simulator version2):** NS2 is a discrete event simulator targeted at networking research. It provides support for simulation of TCP, routing, and multicast protocols over all networks (wired and wireless).

**NS3 (Network Simulator version3):** NS3 is also an open sourced discrete-event network simulator which targets primarily for research and educational use. NS3 is licensed under the GNU GPLv2 license, and is available for research and development. [3]

**OMNET++ (Optical Micro-Networks Plus Plus):** It is an extensible, modular, component-based C++ simulation library and framework, primarily for building network simulators.

## II. BASIC CONCEPTS IN NETWORK SIMULATION

In this section we are going to introduce some basic concepts related to network simulator. We also try to differentiate some confusing key factors of network simulator which create ambiguity among reader.

Generally the basic idea is to clarify of the network simulation and simulator, simulator and then describe the difference between simulation and emulation.

### A. Network Simulation and Simulator

With the help of network simulation any network can be modeled easily that existing in real world. The basic theme of this model is to observe the corresponding results for analysing and the features of the model can be changed. Generally speaking, it's very easy to analyse different scenarios at very low cost and process of model modification is relatively cheap instead of complete real implementation.

However, network simulators are not perfect. They cannot perfectly model all the details of the networks. However, if well modeled, they will be close enough so as to give the researcher a meaningful insight into the network under test, and how changes will affect its operation [4].

### B. Simulation and Emulation

In the area of computer and communications networks research, the most useful technique is simulation that can able to model complete behavior of a network and also able to calculate the interaction between the different network components such as end host, routers, packets or physical links, using different mathematical formulas. Experimental observations from real networks can be handled by simulation and also they can be modeled by virtually or actually capturing. After the observation of data that collects from simulation experiments, a series of offline test experiments implements to check and analyze the supported protocols and behavior of the network. In offline test experiment the simulation allows to modify all kind of environmental

attributes in a controlled manner to evaluate how the network can behave under different parameters, combinations or different configuration conditions. The network simulations have the ability to observe point-to-point or other end-to-end performance in the networks, this is the worth noticing characteristic that the simulation program can be used together with different applications and services.

However, the network emulation means to simulate a network that is under planning in order to evaluate its performance or to forecasts the impression of possible changes, or optimizations. The main points which create difference between them is that end-systems are known as emulator such as in the emulation computers can attached to the emulator and will work same as they attached to the real network. The major point is that the network emulator's job is to emulate the network which connects end-hosts, but not the end-hosts themselves. NS-2 is basically a network simulation tool, which also includes in network emulation tools having limited functionality of emulator, in contrast with a typical network emulator that utilizes some Linux functionality such as WANSim [5] is simple bridged WAN emulator.

### C. Types of Network Simulators

Network simulators further categorized into different type on the basis of some criteria such as they are free or commercial, and they are simple ones or complex ones.

#### 1) Open Source and Commercial Simulators:

Some of the network simulators are free or simply open source which means that they would provide the source code to its user for free. They provide the source code and the affiliated packages which is the main advantage of open source network simulators. In these simulators everything is very open and everyone can contribute to it and find bugs in it. There are no limitations on its interface, so it's also open for future improvements. Cause of its flexibility, new and recent developments of new technologies are implemented in very fast way than commercial network simulators.

On the other hand, the commercial network simulators mean that they would not provide the source code of its software and the affiliated packages to its users for free of cost. To get its licensed version users must have to pay for whole software or they have to pay for specific packages that fulfil their specific usage requirements. Optimized Network Engineering Tool (OPNET) [6] is the main example of commercial simulator. Commercial simulator has its advantages and disadvantages. The main advantage is that it managed and maintained consistently by the some specialized staff of the company and it generally has complete and updated documentations. However, open source network simulator is disadvantageous in this aspect because there are not enough specialized people working on its documentation. This creates lot of serious problems when different versions come with many new things and it will become more difficult to trace or understand the previous codes without any proper documentation.

We can see that commercial network simulators have some advantages, however, open source network simulators some disadvantages, such as, lack of complete and systematic documentations, lack of version control supports can lead to some serious problems and can applicability and life time of the open source network simulators. Typical open source network simulators include NS-2 [7, 8], NS-3 [9]. We will introduce and analyze them in detail in the following sections.

TABLE I  
NETWORK SIMULATORS

Category	Network Simulator Name
Commercial	OPNET, QualNet
Open Source	NS2, NS3, OMNET++, SSFNet, J-Sim

Note that limited by the length, not all of them will be introduced in detail in this paper. However, we will focus on some typical ones and introduce others briefly.

2) *Simple vs. Complex:*

There are lots of network simulators available in the market of different varieties now days. The varieties of network simulators are ranging from the simple ones to the complex ones. Basically any network simulator should enable users to represent a network topology, defining different scenarios, specifying nodes on the network, link between those nodes and the traffic between the nodes. The users are allowed to specify everything about the protocols that used to process network traffic in complicated systems. The graphical applications also allow users to easily visualize the working of their simulated environment. Some of them are providing text-based environment, which has limitations to visualize or intuitive interface, but allow the customization in more advanced forms. Others may be providing a programming framework that customized to create an application that test the networking environment (programming oriented tools).

III. A REVIEW OF EXISTING NETWORK SIMULATORS

While various simulators exist for designing a variety of network models, here in this paper we compare 4 popular network simulators highlighting merits and demerits, strengths and weaknesses, main features, recent developments and their future. These simulators were selected based on their popularity.

A. *OPNET*

Optimized Network Engineering Tool (OPNET) is a commercial network simulator package which is available for supporting both the teaching and research in educational institutes under OPNET university academic program [9]. OPNET is object-oriented, discrete event, general purpose network simulator that provide comprehensive development environment for specification, simulation and performance analysis of computer and data communication networks.

OPNET has different type of modules and tools that increase its development capabilities, including OPNET modeler, model library, planner and different analysis tools [10]. The network industries widely used this simulation tool for observing performance, on the basis of evaluation of local and wide area networks and performance modeling. OPNET is very expensive package, because it has very powerful capabilities for developing different scenarios. Generally it's a comprehensive tool for simulation that includes model library, modular model development, user friendly GUI, high level of modeling detail and customizable presentation of simulation results. The license maintenance fees are very high and its parameter categorization is not very transparent.

The mechanism followed by OPNET is known as discrete event system which means that the system behavior can simulate be modeling the events in the system in such a same order of scenarios as the user has setup. To organize the network it used graded or classified structure.

OPNET is programing oriented tool that provide programming tool for users to define the packet layout of the protocol. The state transition machine, defining network model and the process module can be accomplished with the help of these programming tools. The GUI interface and the programming tools are very useful to help user to build the system as they want, these features make the OPNET simulator popular and very useful in industry for network research and development. The main architecture of OPNET [4] is

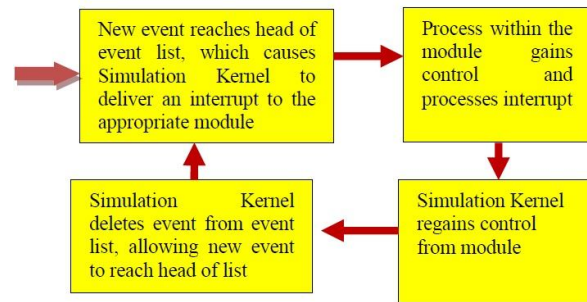


Fig. 1 Architecture of OPNET [4]

1) *Main Features:*

Generally the OPNET can be able to perform three main functions such as modeling, simulating and analysis. It provide very simple but comprehensive graphical environment for modeling, modeling means to create all kind of models of protocols. Further for the simulating it use three different but advanced simulations technologies and also can be used to address a wide range of studies. For analysis, the displayed data from simulations results can be analyzed very easily. The accessibility to the simulator development environment is very ease cause of its user friendly graphs, charts; statistics as well as the animations can be generated by OPNET. Simply OPNET is very convenient for its users.

- a) According to the white paper of OPNET, the detailed features are as follows:
- b) Source code provide lots of components with its library
- c) Fast and discrete event simulation engine
- d) Modeling based on objects (Object Oriented Modeling)
- e) Graphical user interface supports 32-bit and 64-bit
- f) 32-bit and 64-bit parallel simulation kernel
- g) Graded or ordered modeling environment
- h) Wireless modeling also customizable
- i) Scalable wireless simulation support
- j) Also support grid computing
- k) Hybrid, discrete event and analytical simulation
- l) Open interface for integrating external component libraries.
- m) Comprehensive development environment
- n) GUI-based debugging
- o) Both behavior and performance of modeled systems can be analyzed by performing discrete event simulations

2) *Recent Development and its Future :*

OPNET Technologies announced the addition of two major application performance management capabilities, recently, about at August 7, 2008. These capabilities include end-to-end visibility into application performance for organizations using WAN optimization solutions and the ability to capture and analyze NetFlow data.

OPNET, ACE Analyst recently upgrade software includes the functionality and claimed he let the end user organizations using Riverbed, Cisco, Juniper and WAN optimization technology to support end-to-end visibility into application performance when deploying WAN acceleration solutions.

*B. Network Simulator 2 (NS-2):*

The one of the most widely used network simulators and object-oriented discrete-event network simulator is NS-2 that was originally developed at Lawrence Berkeley Laboratory at the University of California, Berkeley, as part of the Virtual Inter Network Testbed (VINT) project [7]. Basically it was designed for network research community to simulate routing algorithms, TCP/IP protocols and multicast. Further it was extended to support node mobility also by the Monarch project at Carnegie Mellon University [10]. It is written in C++ and uses OTcl as a command and configuration interface.

It spends no cost for downloading that is its main strength for Authors. Allowing other researchers to build upon their work using the original code is its strength. From academia point of view, specifically Master's and Doctoral studies it is very useful. The main weakness of ns-2 is the lack of graphical presentations of simulation output data, to process data into suitable format for tools like Xgraph or Gnuplot the raw data must be processed using scripting languages such as 'awk' or 'perl' [7]. Also; it is not a user friendly because of its text based interface. Many student researchers point out that ns-2 has a sharp learning curve. A tutorial contributed by Marc Greis [11] and the continuing evolution of ns documentation have made it easy to learn, but ns-2's split programming model still a barrier to many developers.

1) *Main Features:*

First and foremost, NS2 is an object-oriented, discrete event driven network simulator which was originally developed at University of California-Berkely. The programming it uses is C++ and OTcl (Tcl script language with Object-oriented extensions developed at MIT). The usage of these two programming language has its reason. The biggest reason is due to the internal characteristics of these two languages. C++ is efficient to implement a design but it is not very easy to be visual and graphically shown. It's not easy to modify and assembly different components and to change different parameters without a very visual and easy-to-use descriptive language. Moreover, for efficiency reason, NS2 separates control path implementations from the data path implementation. The event scheduler and the basic network component objects in the data path are written and compiled using C++ to reduce packet and event processing time. OTcl happens to have the feature that C++ lacks. So the combination of these two languages proves to be very effective. C++ is used to implement the detailed protocol and OTcl is used for users to control the simulation scenario and schedule the events. A simplified user's view of NS2 is shown in figure 2. The OTcl script is used to initiate the event scheduler, set up the network topology, and tell traffic source when to start and stop sending packets through event scheduler. The scenes can be changed easily by programming in the OTcl script. When a user wants to make a new network object, he can either write the new object or assemble a compound object from the existing object library, and plumb the data path through the object. This plumbing makes NS2 very powerful.

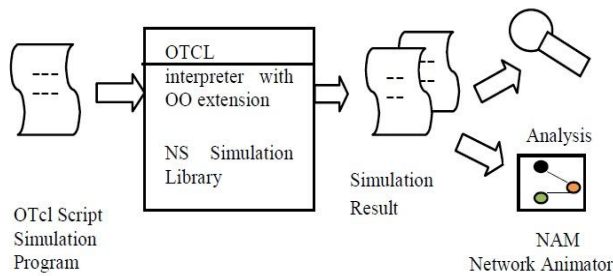


Fig. 2 Simplified User's View of NS-2

Another feature of NS2 is the event scheduler. In NS2, the event scheduler keeps track of simulation time and release all the events in the event queue by invoking appropriate network components. All the network components use the event scheduler by issuing an event for the packet and waiting for the event to be released before doing further action on the packet.

2) *Recent Development and its Future :*

The most recent version of NS2 is NS 2.33 version which was released on Mar 31, 2008. Compared with the previous version, this newest version [7] has integrated the most recent extension on new 802.11 models which include the Ilango Purushothaman's infrastructure mode extensions, the 802.11Ext models from a Mercedes-Benz R&D, NA and University of Karlsruhe team, and the dynamic libraries patch and multirate 802.11 library from Nicola Baldo and Federico Maguolo of the SIGNET group, University of Padova. NS is now developed in collaboration between some different researchers and institutions, including SAMAN (supported by DARPA), CONSER (through the NSF), and ICIR (formerly ACIRI). Contributions have also come from Sun Microsystems and the UCB and Carnegie Mellon Monarch projects. Generation 3 of NS (NS3) has begun development as of July 1, 2006 and is projected to take four years. It is deemed as the future of NS2, and we will discuss the new generation NS 3 in detail in the following section.

C. *Network Simulator 3 (NS-3):*

Similar to NS2, NS3 is also an open sourced discrete-event network simulator which targets primarily for research and educational use. NS3 is licensed under the GNU GPLv2 license, and is available for research and development.

1) *Overview :*

NS3 is a new simulator; it is designed to replace the current popular NS2. It is not an updated version of NS2 and also not backward-compatible with NS2.

2) *Main Features :*

The basic idea of NS3 comes from several different network simulators including NS2, YANS [12], and

GTNetS [13]. The major difference lying between NS3 and NS2 includes:

- a) *Different Software Core:* The core of NS3 is written in C++ and with Python scripting interface (compared with OTcl in NS2). Several advanced C++ design patterns are also used.
- b) *Attention to Realism:* Protocol entities are designed to be closer to real computers.
- c) *Software Integration:* Support the incorporation of more open-source networking software and reduce the need to rewrite models for simulation;
- d) *Support for Virtualization:* Lightweight virtual machines are used. Figure 3 gives an example virtualization testbed of NS3.

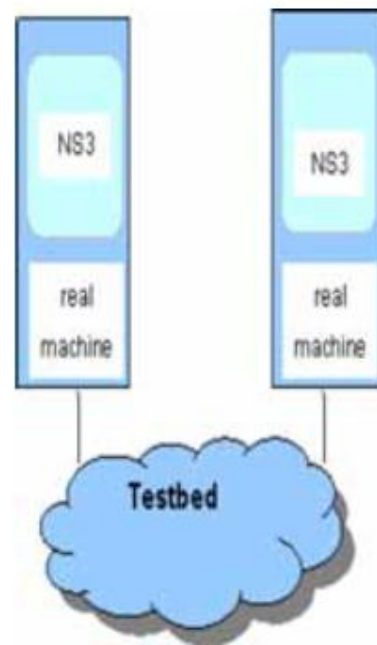


Fig 3 Testbeds interconnect NS3 stacks

- e) *Tracing Architecture:* NS3 is developing a tracing and statistics gathering framework trying to enable customization of the output without rebuilding the simulation core.

Through the comparison between NS2 and NS3, we can summarize the NS 3's features as follows:

1. Modular, documented core
2. C++ programs and Python scripting
3. Alignment with real systems
4. Software integration
5. Virtualization and testbed integration
6. Attribute system
7. Updated models



- i) Utilities (random number seed generation tool, makefile creation tool, etc )
- j) Documentation, sample simulations, etc

To create simulation components, the simulation kernel C++ class library consists of the simulation kernel and utility classes , its library also includes the infrastructure to assemble simulations from different components. Besides these, there are also runtime user interfaces for simulations, and there are also some tools to facilitate and manage simulations. It can run on Unix-like systems like Linux and on Windows (XP, Win2K).

It represents a framework approach as it provides an infrastructure for writing different simulations. There are a lot of open sourced simulation models and frameworks. Another important feature of OMNeT++ is these models are developed completely independently, and follow their own release cycles.

### 3) *Recent Development and its Future :*

Currently, OMNeT++ is popular in academia for its extensibility since it is also open sourced and there are plentiful online documentations. There is also a mailing list for the general discussion. OMNeT++ is being used in the academia as well as in industry. Several open source simulation models have been published in the field of network simulations such as IP, IPv6, MPLS, mobility and ad-hoc simulations.

For the future of OMNeT++, we need to note that OMNeT++ is not a network simulator itself. Actually it is currently popular as a network simulation platform in the academia as well as in industry, and build up a large user community. So we have the reason to believe that using OMNeT++ as a basic platform but not an overall single solution. OMNeT++ can have greater development if it could persuade more organizations to participate in and to contribute.

### IV. SUMMARY

This paper is to offer general overview and a good reference source on the current development status of network

simulators for the people who are not very familiar with this topic, or for someone who want to get some general information related to it. 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 new developments, advantages, disadvantages and their performance, improving on network simulation methodologies.

So in this paper, we first gave a brief introduction on some key concepts of network simulation and network simulators. After that, we introduced four typical network simulators: OPNET, NS2, NS3, and OMNeT++. Their main features, current status and future development are also analyzed and discussed.

In conclusion, the question of which simulator to use is a difficult one and the answer is largely dependent on the specific use case. However, if scalability is the main concern, ns-3 and OMNeT++ are smart choices.

### REFERENCES

- [1] Network Simulator Datasheet: <http://www.webnms.com/simulator/network-simulator-ds.html>.
- [2] [http://en.wikipedia.org/wiki/Network\\_simulation](http://en.wikipedia.org/wiki/Network_simulation).
- [3] Pan, J. and R. Jain, A survey of network simulation tools: Current status and future developments. Email: jp10@cse.wustl.edu, 2008.
- [4] Siraj, M.S., A. Gupta, and M. Rinku-Badgujar, Network Simulation Tools Survey. International Journal of Advanced Research in Computer and Communication Engineering (IJARCCCE 2012), 2012. 1: p. 201-210.
- [5] WAN simulators and emulators : <http://www.wan-sim.net/> .
- [6] OPNET Modeler : <http://www.opnet.com/> .
- [7] Fall, K. and K. Varadhan, The ns manual. The VINT project, from <http://www.isi.edu/nsnam/ns/doc/> . 2013.
- [8] NS2 resource webpage : [http://nsnam.isi.edu/nsnam/index.php/Main\\_Page](http://nsnam.isi.edu/nsnam/index.php/Main_Page)
- [9] NS3 official website: <http://www.nsnam.org/documentation/>
- [10] CMU Monarch Project. Retrieved from: <http://www.monarch.cs.cmu.edu/>
- [11] Greis, M., NS-2 tutorial. Retrieved from: <http://www.isi.edu/nsnam/ns/tutorial/index.html>.
- [12] Yet Another Network Simulator [YANS], <http://yans.inria.fr/code/yans/?summary>.
- [13] The Georgia Tech Network Simulator (GTNetS) , <http://www.ece.gatech.edu/research/labs/MANIACS/GTNetS/>.
- [14] OMNeT++ official website, <http://www.omnetpp.org/>.