A Comparative Study of Various Network Simulation Tools

Arvind T,
Assistant Professor(c),
Dept. Of Computer Science, University College of Science,
Saifabad, Osmania University,Hyderabad.
E-mail: mr.arvind@rediffmail.com

Abstract—In the area of network research, establishing a network in a real time scenario is very difficult. A single test bed takes a large amount of time and cost. Thus a network simulator tool helps the network developer to check whether the network is able to work in the real time. A simulator also saves a lot of time, money and makes the implementations easy. In this paper, we introduce the main features of different network simulator and consider their advantages and disadvantages. We hope this paper proves to be a good reference source for those people who feel difficult to select the appropriate network simulators for their research.

Keywords: Network Simulator, NS-2, NS-3, OPNET, OMNeT++, J-Sim and QualNet.

I. INTRODUCTION

Network simulation is the most useful and common methodology used to evaluate different network topologies without real world implementation. Network simulation is used in different areas, academic researchers, industrial development, to analyse, design, simulate and verify the performance of different network theories and hypotheses. There are a number of network simulators for instance, ns-2, ns-3, OMNET++, OPNET, Jism, QualNet and GloMoSiM etc.The main focus of this paper is to assist researchers in choosing the most suitable simulators for their research work.

II. INTRODUCTION OF NETWORK SIMULATORS

A. NS-2

Network Simulator2 (ns-2) is an open source,discrete event network simulator. It is used for the simulation of network protocols with different network topologies. It is capable of simulating wired as well as wireless networks. NS-2 was built in C++ and provides the simulation interface through OTcl, an object-oriented dialect of Tcl.

The user describes a network topology by writing OTcl scripts, and then the main NS program simulates that topology with specified parameters. In ns-2, network animator (NAM) is used for the graphical view of the network. ns-2 is the most common and widely used network simulator for research work. NAM interface contains control features that allow users to forward, pause, stop and play the simulation.

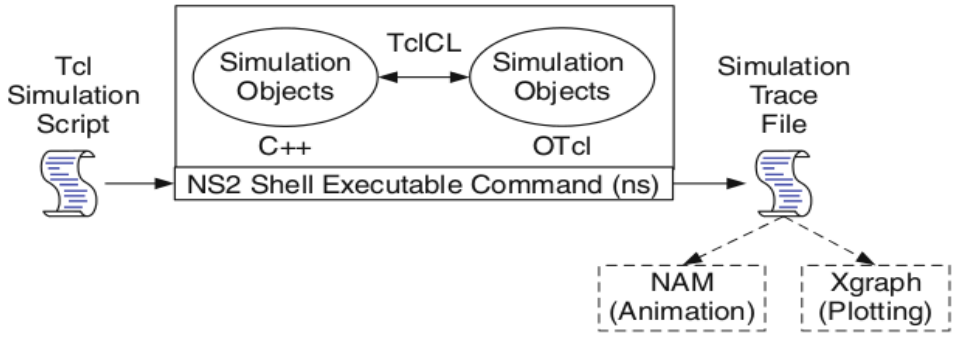

Fig 1. Basic architecture of ns-2

B. NS-3

The ns-3 simulator is a discrete-event network simulator for Internet systems, targeted primarily for research and educational use. The ns-3 project, started in 2006, is an open-source project developing ns-3. Ns-3 is free software,licensed under the GNU GPLv2 license. It will rely on the ongoing contributions of the community to develop new models, debug or maintain existing ones, and share results.

Table 1. Comparison of ns-2 and ns-3

Criteria	**Ns-2**	**Ns-3**
Released at	1996	2008
Supported by	DARPA, VINT, SAMAN,NSF & INRIA	NSF & CONSER
Based on	NS-1 & REAL simulator	NS-2, GTNets, YANS
Built in	C++	C++
Scripting	OTcl	Python
Simulation output	NAM	NS-3-viz,pyviz,nam

Programming Languages: Ns-2 is implemented using a combination of oTCL (object Tool Command Language) and C++ while Ns-3 is implemented entirely in C++ and support of python for scripting and visualization.

i. **Memory Management:** Ns-2 requires basic manual C++ memory management function. Because Ns3 is implemented in C++, all normal C++ memory management functions are available.

ii. **Performance:** Ns3 performance is better than Ns2.

iii. **Simulation outcome:** Ns-2 come with the package NAM (Network Animator) it's a tcl based animation system.

In Ns-3 having python based visualization package Furthermore, Ns3 don't have all the modules that ns-2 has, but some modules are ported from ns-2 to ns3.

C. OPNET (Optimum Network Performance)

This simulator is developed by OPNET technologies; Inc. OPNET had been originally developed at the Massachusetts Institute of Technology (MIT) and since 1987 has become commercial software. OPNET is a high level event based network level simulation tool •Simulation operates at "packet-level".It provides a comprehensive development environment supporting the modeling of communication networks and distributed systems. Both behavior and performance of modeled systems can be analyzed by performing discrete event simulations.

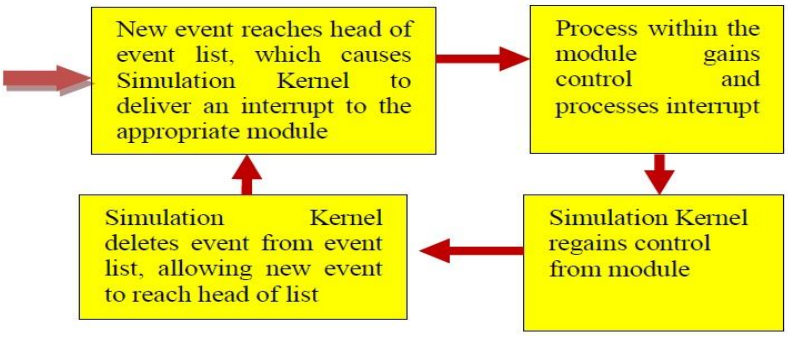

Fig 2. Basic architecture of OPNET

D. OMNET++ (Optical Micro-Networks Plus Plus)

It is a component-based, modular and open-architecture discrete event simulator framework. The most common use of OMNeT++ is for simulation of computer networks, but it is also used for queuing network simulations and other areas as well. It is licensed under its own Academic Public License, which allows GNU Public License-like freedom but only in non-commercial settings. OMNET++ distributions are available for both UNIX and Windows-based systems. It was developed using component-oriented approach that promotes structured and reusable models. In addition, OMNET++ has extensive graphical user interface (GUI) and intelligence support.

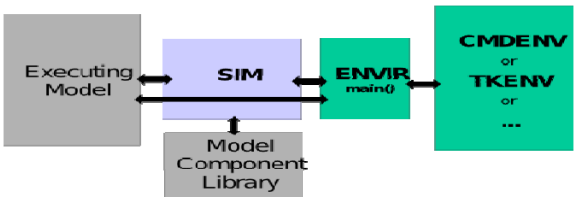

Fig 3. Basic architecture of OMNeT++

E. JSIM

It is a component-based compositional simulation environment based on the autonomous component architecture (ACA). The basic entities of ACA are components,which communicated with each other by sending and receiving data using their ports. The components' behaviour is specified at design time in contracts – their actual binding will linked at system integration time. The loosely-coupled component architecture of JSim enables the user to design, implement and test single components individually. New components can be easily added or exchanged for existing ones. On top of ACA a generalised packet switched network model, the so called Internetworking Simulation Platform (INET), is developed tha The combination of JSim as Java implementation and the component-based architecture makes JSim a platform-independent, extensible and reusable simulation environment.Moreover, JSim provides a scripting interface that enables the use of script languages such as Perl, Tcl and Python.

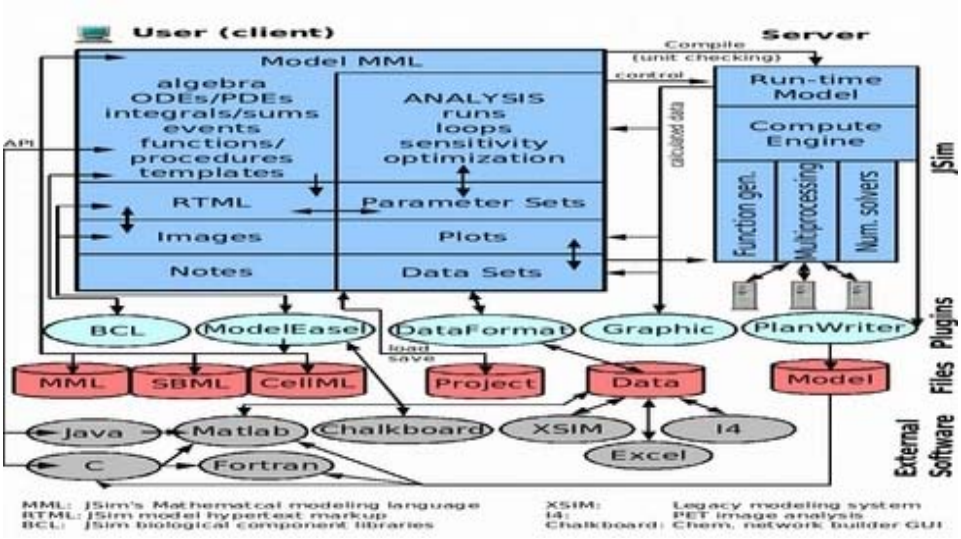

Fig 4. Jsim architecture

F. QualNet

QualNet is a state-of-the-art simulator for large, heterogeneous networks and the distributed applications that execute on those networks.QualNet is a commercial version of GloMoSim that runs on all common platforms (Linux, Windows, Solaris, OS X) and is specialized in simulating all kind of wireless applications..It is ultra high-fidelity network simulation software that predicts wireless, wired and mixed-platform network and networking device performance.

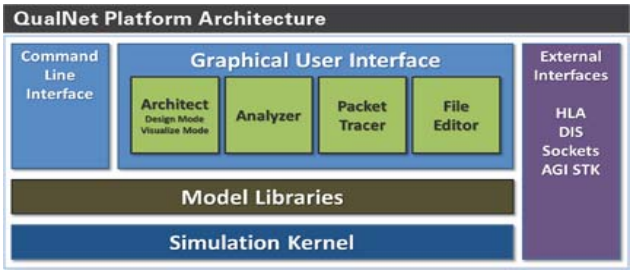

Fig 5. Architecture of QualNet

III. COMPARSION BETWEEN SIMULATION TOOLS

Table 2. Comparison of different Network Simulation tools

Features	NS2	NS3	JSIM	OMnet++	OPNET	QualNet
Language Supported	C++/OTCL	C++,Python	Java,Tcl	C++	C (C++)	Parsec C++
GUI support	Poor	Good	Good	Good	excellent	Excellent
Time taken to learn	Long	Moderate	Moderate	Moderate	long	Very easy
Time take for download and installation	Moderate time	Long time to download and install all necessary patches and supporting software.	Easy to download and install	Very easy takes very less time. Easily available	Moderate time	Easy to download and install
Platform	Linux, unix,windows, Cygwin	Linux, unix, windows	windows, linux, Matlab	Linux,unix, windows, MAC OS	Linux,solaris windows	Linux, DOS Windows
Network Visualization tool	It supports network visualization tool	It supports network visualization tool	It supports network visualization tool	It supports network visualization tool	It supports network visualization tool	It supports network visualization tool
Availability of analysis tool	It has analysis tool	It has analysis tool	It has analysis tool	It has analysis tool	It has analysis tool	It has analysis tool
Crates trace file	It creates trace file	It creates trace file	It creates trace file	It creates trace file	It creates trace file	It creates trace file
Possibility to design and modify the network scenarios	It is Possible	It is Possible	It is Possible	It is Possible	It is Possible	It is Possible
Design and Implementation Protocols	Supports both wired and wireless simulation of protocols	Supports both wired and wireless simulation of protocols	Supports both wired and wireless simulation of protocols	Supports both wired and wireless simulation of protocols	Supports both wired and wireless simulation of protocols	Supports both wired and wireless simulation of protocols
Interaction with real time system	It is Possible	It is Possible	It is Possible	It is Possible	It is Possible	It is Possible
Fast simulation capabilities	Moderate	Moderate	Poor	Moderate	excellent	excellent
Merits	-Easy to add new protocols. -A large number of protocols available publicly.	-NS-3 is not an extension of NS-2 it is a new simulator. -NS-3 is open-source -Support for virtualization	-Provides support for energy modelling, with the exception of radio energy consumption - Support mobile wireless networks and sensor networks. -Component-oriented architecture	-Powerful graphical User Interface (making tracing and bugging easier) -Simulate power Consumption problem	-Opnet communicates with other simulators -Fast discrete event simulation engine -Scalable wireless simulation support	-Powerful graphical User Interface -It shows a very good scalability, simulation time being reasonable. -Qualnet can support real-time speed to

					-Integrated, GUI-based debugging and analysis	enable software-in-the-loop, network emulation, and hardware-in-the-loop modelling
De merits	-Supports only two wireless MAC protocols, 802.11, and a single-hop TDMA protocol. -Need to familiar with writing scripting language	-Python bindings do not work on Cygwin. -Only IPv4 is supported.	-Low efficiency of simulation. -The only MAC protocol provided for wireless networks is 802.11. -Unnecessary run-time overhead	-Number of protocol is not large enough. -Compatibility problem (not portable) -OMNeT++ is a bit slow due to its long simulation run and high memory consumption	-It is a commercial product -Memory consuming models -insufficient tutorials	-Qualnet is a Commercial Product -Difficult installation on Linux.

IV. CONCLUSION

In this paper we have investigated different network simulating tools and found that Analysis tool for state estimation is available in Qualnet, Omnet++, Opnet, Ns2, Ns3 and J-Sim. All the simulators discussed in the paper supports network visualization tools and can modify scenarios. Trace files are created by all the simulators discussed in the paper. Qulanet and Opnet communicates with other simulators whereas Omnet++, Ns-2 and J-Sim do not. Fast simulation capabilities are only supported by Qualnet and Opnet. Ns-2 consumes more amount of memory and Ns-3 consumes lowest amount of memory when compared to other simulator as the number of nodes increases.Qualnet and Omnet++ are commercial simulators and are capable to simulate larger networks, whereas Ns-2, Ns-3, Omnet++ and J-Sim are open source.

REFERENCES

[1] Jianli Pan, Prof. Raj Jain,Project , “A Survey of Network Simulation Tools: Current Status and Future Developments”,report.
[2] Mrs. Saba Siraj, Mr. Ajay Kumar Gupta, Mrs Rinku Badgujar, “Network Simulation Tools Survey”,
[3] International Journal of Advanced Research in Computer and Communication Engineering Vol. 1,Issue 4, June 2012, ISSN :2278 – 1021.
[4] Atta ur Rehman Khana, Sardar M. Bilalb, Mazliza Othmana ,” A Performance Comparison of Network Simulators for Wireless Network”.
[5] Mrs.Uma R Pujeri,Dr.. V Palanisamy,”Survey of Various Open Source Network Simulators”,
[6] http://en.wikipedia.org/wiki/Network_simulation
[7] The NS-3 network simulator :http://www.nsnam.org/
[8] Analysis and comparison of different network simulators Vinita Mishra1 , Smita Jangale 2
[9] [OMNeT] OMNeT++ official website, ttp://www.omnetpp.org/
[10] E.Weingartner,H.V.Lehn and k.wehrle “A performance comparison of recent network simulator” IEEE International Conference on Communications (ICC 2009), RWTH Aachen University Aachen, Germany pp. 271–350
[11] J-Sim. http://www.j-sim.org/.
[12] ns-3 Overview. *https://www.nsnam.org/docs/ns-3-overview-july-2014.pdf*
[13] OPNET *www.sce.carleton.ca/faculty/lambadaris/courses/5001/opnet_tutorial.pdf*