



New Innovations in Performance Analysis of Computer Networks: A Review

A. A. Obiniyi

Department of Mathematics,
Ahmadu Bello University, Zaria

M. B. Soroyewun

School of Science &
Technology, National Open
University of Nigeria, Kaduna
Study Center, Kaduna

M. M. Abur

Iya Abubakar Computer Center,
Ahmadu Bello University, Zaria

ABSTRACT

Innovation is the application of new solutions that meet new requirements, inarticulate needs, or existing market needs. In the field of performance analysis of networks, new innovations are springing up which are focusing on converged networks (data, voice and video). Among the recent ones are both open source and commercial tools developed to improve and simplify performance analysis techniques vis-à-vis analytical modeling, simulation and measurements. In this paper, an attempt is made to consider the concept of performance analysis in general while focusing on computer networks. The three broad techniques currently being employed in performance analysis of computer networks are explored. The nature and the requisite skills for a performance analyst are enumerated. In addition, the performance metrics of computer networks are reviewed with available software tools for carrying out network performance analysis. Finally, the new innovations and future focus of performance analysis with emphasis on computer networks with a view to intimating new researchers in this field with requisite knowledge for a less stressful research endeavour is considered.

General Terms

Innovations, Performance Analysis, Computer Networks, Modeling, Simulation, Analytical Modeling, Simulation, Measurements, Techniques

Keywords

Performance Analysis, Markov Chains, PEPA, Stochastic Petri Nets, Simulation Models, Vehicular Network Simulator

1. INTRODUCTION

Performance analysis of scientific phenomena has been attracting increased research efforts for some time now. This is due to the prospects of uncovering hidden details made possible by it. Performance analysis uncovers several perspectives on a problem or phenomenon by determining any and all drivers towards or barriers to successful performance and proposing solution system based on what is discovered [1].

A computer network essentially consists of an interconnection of devices such as computers, printers, routers and switches that exchange information with each other [2]. Basically, there are three broad techniques or methodologies employed for the performance analysis of computer networks; namely, analytical/mathematical modeling, simulation and measurements [3]. In this paper, new trends and recent

advances in the application of these techniques in performance analysis of computer networks was explored. This will intimate new researchers in this field with required knowledge for a less stressful research ventures.

2. OVERVIEW OF PERFORMANCE ANALYSIS OF COMPUTER NETWORK

With the rapid increase in computer network usage for various applications, its performance analysis or evaluation is becoming increasingly important in order to understand performance issues with a view to proffering new solutions and improvement to network design issues. Computer network users, administrators, and designers are all interested in performance evaluation since their goal is to obtain or provide the highest performance at the lowest cost. Performance analysis is required at every stage in the life cycle of a computer system, including its design, manufacturing, sales, use and upgrade [4]. The same is applicable to computer network as a system.

According to [4], performance analysis is an art which must be learnt, developed and honed, contrary to common belief. It was put forward that like a work of art, successful analysis cannot be produced mechanically since every evaluation requires an intimate knowledge of the system being modeled or analyzed and a careful selection of the technique, workload, parameters and tools. Hence, the major part of performance analyst's job as an art, normally begins with the definition of the real problem, transforming it into a form in which established tools and techniques can be employed while taking in to consideration time and other necessary constraints.

In effect, performance analysts should generally belong to at least two different types of professional organizations; one dealing with performance and the other dealing with the type of systems of their interest. For instance, a network system performance analyst would belong to network organizations and performance organizations. There are a number of organizations that are exclusively devoted to performance analysis according to [4] and they are listed as follows:

- a. ACM SIGMETRICS – the Association for Computing Machinery's Special Interest Group concerned with computer system performance analysis.
- b. IEEE Computer Society – Institute of Electrical and Electronics Engineers (IEEE) Computer Society has a technical committee on simulation.



- c. ACM SIGSIM – The ACM Special Interest Group on Simulation.
- d. The Society for Computer Simulation
- e. The Computer Measurement Group (CMG)
- f. The Society for Industrial and Applied Mathematics (SIAM)
- g. The International Federation for Information Processing (IFIP) Working Group 7.3
- h. The Operations Research Society of America (ORSA)

Also of relevance to the analyst is the availability of journals primarily devoted to computer systems performance evaluation from the above listed (a-h) performance organizations. Moreover, courses on statistical inference, operations research, stochastic processes, decision theory, time series analysis, design of experiments, system simulation, queuing theory and other related subjects may be considered in order to enhance the skill of the performance analyst [4].

In order to avoid common mistakes usually made in performance analysis, a comprehensive list of the systematic approach to performance evaluation were detailed in [4] which are summarized as follows:

- a. Stating the goals and defining the system boundaries
- b. Listing the services and possible outcomes of the system
- c. Selecting the correct performance metrics related to the system
- d. Listing the system and workload parameters
- e. Selecting the factors to study and their values
- f. Selecting the right evaluation techniques for the study
- g. Selecting the workload that is representative of the real life usage of the system
- h. Designing the experiments for the study
- i. Analyzing and interpreting the data
- j. Presenting the results in an easily understandable manner.

2.1 Reasons for Performance Study

There are a number of reasons performance analysis study may be undertaken as stated in [7]. They are mostly economic reasons used in taking investment decisions and are summarized in *a* to *e*.

- a. To find out performance bottlenecks in existing systems and develop improvements.
- b. For capacity planning: for instance, how much resources should be spent to obtain some desired level of service quality?
- c. For performance comparison of systems, algorithms and protocols; for example, given two protocols which one is better and in which respect?
- d. To verify the claims of product designers and manufacturers or service providers; e.g. is the Internet service provider able to guarantee a certain minimum bandwidth as promised?
- e. For predicting the performance at future workloads or operating conditions (forecasting).

2.2 Performance Metrics Used In Computer Networks

According to [8], a performance metric is a standard definition of a measurable quantity that indicates some aspect of performance. Performance metrics are the criteria used to evaluate the performance of a system [7]. There are a number of metrics applicable to computer networks, but the focus of this write-up will be on those metrics which are of greatest relevance to network performance [9]. New innovations are being employed to accurately monitor, measure and study their effects on network performance. The mostly concerned performance metrics include the following [6] [9]:

- a. Availability
- b. Throughput
- c. Packet Loss
- d. Latency
- e. Delay
- f. Jitter
- g. Bandwidth

2.2.1 Availability

Availability metric assesses how robust the network is; it presents the percentage of time the network is running without any failure of specific network elements or any problem affecting the availability of services [9]. It can also include the Mean Time To Failure (MTTF) which means the uptime [10]. This is the time a machine, typically a computer, has been working and available.

2.2.2 Throughput

This refers to the average rate of successful data or message delivery over a communication link or system and is usually measured in bits per second (bit/s or bps) [6].

2.2.3 Packet Loss

Packet loss and error metrics are indicative of the network congestion conditions and/or transmission errors and/or equipment malfunctioning. They basically measure the fraction of packets lost in a network due to buffer overflows or other reasons or the fraction of error bits or packets [9].

2.2.4 Latency

Latency refers to the amount of time it takes for data to travel from one location to another across a network. It is usually measured in millisecond (ms). It is sometimes referred to as delay because software may often wait to execute some function while data travels back and forth across the network [11] [6].

2.2.5 Delay

In a general sense, delay refers to a lapse of time. Delay metrics also assess the network congestion conditions or effect of routing changes. The delay of a network specifies how long it takes for a bit of data to travel across the network from one node or endpoint to another. In Internet Protocol (IP) network, this is generally the round trip delay for an IP packet within the IP network [6][9][15].



2.2.6 Jitter

Jitter is often used as a measure of variability over time of the packet latency across a network. A network with constant latency has no variation (or jitter). Packet jitter is expressed as an average of the deviation from the network mean latency. It is usually referred to as Packet Delay Variation (PDV) [13]. Jitter often refers to variation in the amount of latency [11].

2.2.7 Bandwidth

Bandwidth means the used capacity or available capacity. It may also refer to consumed bandwidth, corresponding to achieved throughput which is the average rate of successful data transfer through a communication path [14]. Bandwidth metrics, in practice, assess the amount of data that a user can transfer through the network in a time unit and may be dependent or independent from the existing network traffic [9].

3. PERFORMANCE ANALYSIS TECHNIQUES USED IN COMPUTER NETWORKS

Performance is one of the most important non-functional aspects of any system (either hardware or software). Performance analysis applies certain techniques to existing or envisioned systems such as computer systems and communication networks to assess performance measures of interest for example delay, response times, throughput, processing times or jitter [7].

The three techniques mostly employed in performance evaluation as stated earlier are analytical (or mathematical) modeling, simulation and measurements [4]. The choice of the techniques used depends on a number of criteria. When the system under investigation already exists as when designing an improved version of a product, measurement technique can be used. On the other hand, when the system does not exist or is too difficult to deal with, a performance model must be developed. Modeling may be an abstract mathematical representation of the system behaviour or an operational abstraction implemented as a simulator [18]. Analytical modeling and simulation models can be used for situations where measurement is not possible. However, a combination of two or more of the techniques would be more convincing to others [4] [7]. Sections 2.2, 2.3 and 2.4 reviewed these techniques.

3.1 Analytical Modeling

A model is an approximate representation of a physical situation. A useful model explains all relevant aspects of a given situation. Such models can be used instead of experiments to answer questions regarding the given situations. When adequately constructed, models therefore allow the analyst to avoid the costs of experimentation namely; labour, equipment and time. Analytical or mathematical models are used when the observational phenomenon has measurable properties [19]. The main existing mathematical techniques are based on the following formalisms: Operational laws, Queuing networks, Petri nets and Process algebras, with some variants like Stochastic Petri Nets and Stochastic Automata Networks. These techniques

are basically special cases of Markov chain or more complex discrete state non-Markovian models [18]. The various techniques that fall under analytical modeling are discussed in this section. However, it is needful to briefly introduce operational laws here. Basically, *operation laws* are simple equations which may be used as an abstract representation or model of the average behaviour of almost any system. Their strength lies in their simplicity and in the fact that they are very general; almost no assumptions are made about the behaviour of the random variables characterizing the system [39].

3.1.1 Markov Chains

A Markov Chain is a collection of random variables $\{X_t\}$ (where the index t runs through $0, 1, \dots$) having the property that, given the present, the future is conditionally independent of the past [20].

Hence,

$$P(X_t = j | X_0 = i_0, X_1 = i_1, \dots, X_{t-1} = i_{t-1}) = P(X_t = j | X_{t-1} = i_{t-1}) \quad \dots \text{(eqn. 1)}$$

Generally, if a Markov sequence of random variates X_n take the discrete values a_1, a_2, \dots, a_n then

$$\begin{aligned} P(x_n = a_{in} | x_{n-1} = a_{in-1}, \dots, x_1 = a_{i1}) = \\ P(x_n = a_{in} | x_{n-1} = a_{in-1}), \quad \dots \text{(eqn. 2)} \end{aligned}$$

and the sequence x_n is called a Markov chain [20].

Discrete time Markov chain (DTMC) or continuous time Markov chain (CTMC) are state transition systems in which each transition has an associated probability (in DTMCs) or rate (in CTMCs). Hence they can be used to model a wide class of concurrent systems that satisfy the Markov property. Semi-Markov chains (SMCs) relax the Markov property assumption in CTMCs by requiring it to hold only at state transition instants. Then the times spent in each state need not be exponential (as in a CTMC) and can depend on the next state as well. In recent time, SMCs have been used to enhance expressive power in performance modeling [18].

3.1.2 Queuing Networks

Queuing theory is the mathematical study of waiting lines or queues. In it, a model is constructed so that queue lengths and waiting times can be predicted. It has found useful applications in operation research, telecommunications, traffic engineering, computing and the design of diverse building structures [21]. A queuing system describes the system as a unique resource while a queuing network describes the system as a set of interacting resources [22]. Queuing Network (QN) also refers to as networks of queues are systems of a number of queues that are connected by customer routing. A QN can be formalized as a directed graph in which the nodes are queues, often called service centres, each representing a resource in the system. Customers representing the jobs, users or tasks in the system, flow through the model and compete for these resources [18] as shown in figure 1.

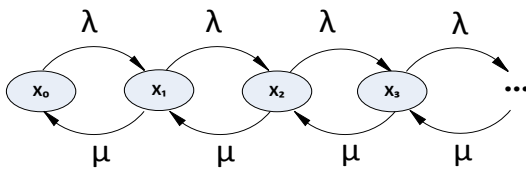


Figure1: The state transition diagram for a simple M/M/1 queue (Source: [23])

The arcs of the network represent the topology of the system, and together with the routing specification determine the paths that customers take through the network [18].

3.1.3 Petri Nets (PN)

Petri nets were originally introduced by C.A. Petri in 1962 and are essentially a graphical and mathematical modeling tool applicable to many systems [24, 25]. According to [24] a PN is a 5-tuple $PN = (P, T, F, W, M)$, where

$P = \{p_1, p_2, \dots, p_m\}$ is a finite set of *places* (drawn as circles);

$T = \{t_1, t_2, \dots, t_n\}$ is a finite set of *transitions* (drawn as bars);

$F \subseteq (P \times T) \cup (T \times P)$ is a set of arcs connecting P and T ;

$W : F \rightarrow \{1, 2, 3, \dots\}$ is weight function;

$M : P \rightarrow \{0, 1, 2, \dots\}$ is the *marking* which denote the number of tokens (drawn as black dots) in place P and the initial marking is denoted as M_0 .

Petri nets are directed graphs with two types of nodes, places and transitions, and unidirectional arcs between them. Generally, tokens move between places according to the firing rules imposed by the transitions [18]. PN have been found to be very useful in the study of behaviour of many real-world scenarios such as concurrency, sequencing, synchronization and conflict. Essentially in computer networks, it can be used to describe and verify the communication protocols [18, 25].

3.1.3.1 Stochastic Petri Nets (SPN)

Since the concept of time is not explicitly stated in the original definition of Petri nets, hence, for performance analysis of dynamic systems, it is important and useful to introduce time delay associated with transitions in the PN models. Such PN model is said to be deterministic (or timed) net or a stochastic net if the delays are deterministic or probabilistic in nature respectively [24, 26]. SPN are Petri nets where exponentially distributed firing time is attached to each transition. A detailed treatment on SPN can be found in [24, 25 and 26].

In Generalized Stochastic Petri nets (GSPN) [27], transitions are allowed to be either *timed* or *immediate*. Immediate transitions always have priority over timed transitions to fire. When SPN is applied to performance analysis of computer networks, places can be used to denote number of packets or cells in the buffer or the number of active users or flows in the system, while the arrival and departure of packets, cells, users or flows can be represented by transitions.

Extended Stochastic Petri Nets (ESPNs) was defined in 1984 by the “Duke group” (Trivedi and his student Bechta Dugan) which put the emphasis on general distributions and transient analysis of SPNs using simulation. The result of their work led to the development of DEEP, a FORTRAN package for the simulation of ESPN [28].

3.1.3.2 Stochastic Reward Nets (SRNs)

SRNs are actually derived from GSPN but extend it further [30]. Every tangible marking can be associated with a reward rate in SRN. It is quite possible to map SRN into Markov reward model giving rise to a variety of performance measures which can then be specified and calculated using a convenient formalism [24]. Other variants or extensions to SRN are Non-Markovian Stochastic Petri Nets and Fluid Stochastic Petri Nets which are treated in details in [24 and 30].

3.1.4 Stochastic Automata Networks

Stochastic Automata Network (SAN) is a formalism for the definition and the solution of complex systems with a very large state space. The main advantage of using SAN is its memory efficiency [31, 32]. Its basic idea is to represent a whole system by a collection of subsystems with an independent behaviour and occasional interdependencies. Each subsystem is described as a stochastic automaton which represents a certain number of states, with rules or probability functions that govern the movements from one state of the automata to the other [33]. SAN has been used efficiently in areas like machine decomposition, systolic computing, neural computing and IP networks [34, 33].

3.1.5 Performance Evaluation Process Algebra (PEPA)

PEPA is a stochastic process algebra designed for modeling computer and communication systems introduced by Jane Hillston in 1994 [36, 37]. In PEPA the actions are assumed to have a duration or delay. Rates are drawn from the exponential distribution and PEPA models are finite-state and so give rise to a stochastic process. Hence it can be used to study quantitative properties of models of computer and communication systems. PEPA has a small set of combinators; prefix, choice, co-operation and hiding. The main advantage of characterizing the corresponding class of PEPA models is that by ‘lifting’ the definition from the stochastic process level in to a formally defined high-level modeling paradigm, automatic detection of these structures is facilitated when they occur [37, 38].

We have endeavored to describe all the available formalisms under mathematical modeling. In the following sections we proceed to discuss the remaining two techniques.

3.2 Simulation Models

Analytical models considered in earlier section, are a representation which can be analyzed mathematically to deduce the behaviour of the system and is often referred to as a *mathematical abstraction* of the system. Simulation models, on the other hand, can be regarded as an *algorithmic*



abstraction of the system that produces a representation which when executed reproduces the behaviour of the system [40].

A simulation of a system is the operation of a model of the system. The model can be reconfigured and experimented with; usually this is not possible or impractical to do in the real-life system. Generally, simulation is a tool to evaluate the performance of a system, existing or proposed under different configurations of interest and over a long period of real time [41, 40]. There is discrete event simulation and continuous simulation. The former is less detailed than the latter but it is much simpler to implement. Hence it is used in a wide variety of situations. Figure 2 shows the basic steps involved in carrying out a simulation study.

A simulation model is a computer program written in a general-purpose or specific simulation language. It usually implements the most important aspects of the system under study in a simplified manner and allows for a greater level of detail than analytical modeling [10].

As stated in [41] the steps involved in developing a simulation model, designing a simulation experiment and performing simulation analysis are: Identify the problem, Formulate the problem, Collect and process real system data, Formulate and develop a model, Validate the model, Document model for future use, Select appropriate experimental design, Establish experimental conditions for runs, Perform simulation runs, Interpret and present results and Recommend further course of action. For detailed discussion on the aforementioned steps, other benefits of simulation modeling and pitfalls to avoid please see [4, 41].

3.3 Empirical Measurement

Measurement-based performance analyses have become an increasingly important component of network research. Measurements are carried out by using software and hardware

monitoring devices. Hardware monitoring devices include probes and sensors, counters, as well as devices for data display and recording. In addition, software monitoring records certain events and information about the system state. This is otherwise called operational analysis which covers measurement and estimation of the real system. Analysis results can then be used for improving performance and setting up new bounds of the system [43].

Measurement is necessary for understanding current system behavior and how new systems will behave. Measurement is meaningless without careful analysis. Measurement and analysis enable models to be built which can be used to effectively develop and evaluate new techniques. It is important to determine what to measure before any measurements take place [44]. Many tools have been developed to measure/monitor general characteristics of network performance. All network performance measurement tools can be categorized into two broad categories: (1) non-intrusive measurement or passive tools e.g. tcpdump and Simple Network Management Protocol (SNMP) tools and (2) intrusive measurement or active tools e.g. traceroute and ping [44, 45].

Non-intrusive or passive measurement measures the network behavior by observing the packet arrival rate at an end system and making some deduction on the state of the network, thereby deducing the network performance on the basis of these observations. In contrast, intrusive or active measurement refers to the controlled injection of packets into the network and the subsequent collection of packets [45].

Invariably, measured data need to be analyzed and the results should be properly interpreted. Analyzing such data in networks is usually done using statistical methods like averaging, dispersion (variability), correlations, regression analysis, distributional analysis, frequency analysis, principal-component analysis and cluster analysis [44].

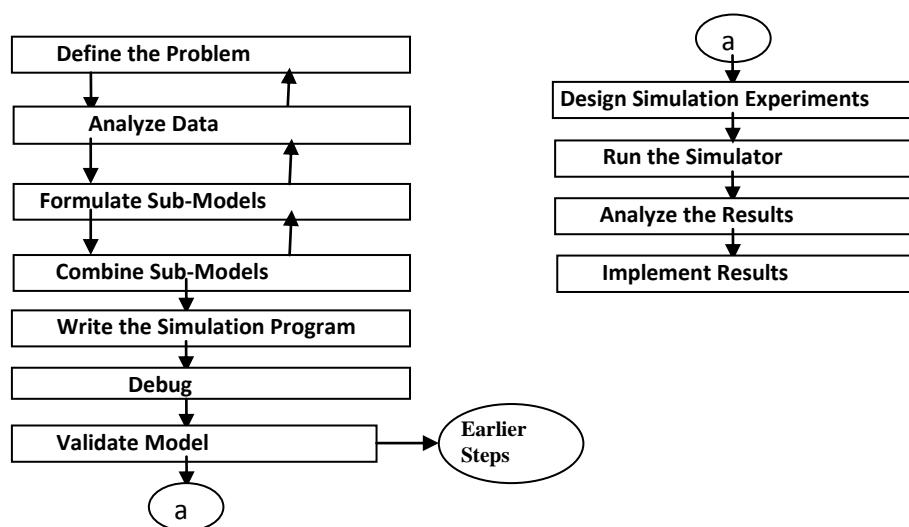


Figure 2: Basic steps involved in a simulation study (Source: [42])



4. AVAILABLE SOFTWARE TOOLS FOR NETWORK PERFORMANCE ANALYSIS

There are a plethora of tools available for the performance analysis of networks. In this section, we consider software tools available for use in each of the major categories; analytical modeling, simulation modeling and empirical measurements.

4.1 Software Tools for Analytical Modeling

Analytical modeling abstracts the features of a computer network system as a set of parameters or parameterized functions in order to make the modeling task easier [46]. These mathematical derivations which give rise to some system equations are then solved to yield the network performance. Considering that computer networks are non-deterministic in behavior or stochastic, statistical models are usually employed to represent the random events happening in computer networks. MATLAB, Mathematica, Maple and their open source counterparts like GNU Octave are available for this purpose. Another tool called PEPA workbench was used in [35] and enunciated in [54]. PEPARoni, a discrete event simulator developed by Graham Clark and PEPA Plug-in Project are available in [55].

4.2 Software Tools for Simulation Modeling

In [47], a brief overview of suitable simulation tools for education and research on network was presented. The ones mentioned in the paper are OPNET (Optimum Performance Network) Modeler, OPNET IT Guru academic edition, Network II.5, SES/Workbench, BONes Designer and COMNET III. In addition, the advanced modeling features of OPNET Modeler were discussed.

NS-3 (Network Simulator -3) and OMNeT++ (Objective Modular Network Testbed in C++) were extensively discussed in [49] as discrete-event based network simulation tools. The book is aimed at graduate students and researchers in the field of network technologies. A comprehensive list of computer simulation software classified under open source and proprietary is available in [48] while [56] did an overview of about 42 different simulation and modeling tools. Under open source, the list contained ASCEND, DWSIM, Elmer Facsimile, Galatac, GNU Octave, NS-3 and SimPy. Of interest in proprietary class are NetSim, Wolfram System Modeler, MATLAB, ExtendSim and HyperWorks.

4.3 Software Tools for Network Measurement and Analysis

A number of software tools are available for network performance monitoring and measurement. They are broadly categorized into non-intrusive and intrusive measurement tools according to [45]. In [50, 51], common non-intrusive data collection tools are tcpdump, tcptrace, Corsan, IPTraf, fsflows, NeTraMet, NetPredict and CoralReef while intrusive or active probing tools are mping, pathcar, treno, traceroute,

echoping, ping, Nettek, Traceping, Imeter, scamper and dummynet among others.

Other tools like Ntop, nfstat, NfSen, JKFlow, Stager, Argus, AutoFocus, CoMo, Cricket Beluga, cflowd, Wireshark, Fluxoscope, Flamingo, NetFlow Monitor and NetPy are described in [51].

5. NEW INNOVATIONS IN PERFORMANCE ANALYSIS OF NETWORKS

Innovation is the application of new solutions that meet new requirements, inarticulate needs, or existing market needs. An innovation is something original, new and important – in whatever field – that breaks into (or obtains a foothold) in a market or society [53]. In the field of performance analysis of networks, new innovations are springing up which are focusing on converged networks (data, voice and video). Among the recent ones are both open source and commercial tools developed to improve and simplify performance analysis techniques vis-à-vis analytical modeling, simulation and measurements. Of particular focus in this write-up are PEPA Plug-in/pypepa, MATLAB, SevOne, WinEyeQ, NS3, OMNeT++, GNS3, VNS and OPNET.

5.1 PEPA Plug-in / pypepa

The PEPA Plug-in project is a software tool that supports the stochastic process algebra PEPA. The tool enables Markovian steady-state as well as ODE analysis of PEPA models, and now has new interface for abstracting and model checking Continuous Stochastic Logic (CSL) properties. It is deployed as a contribution to the Eclipse Framework [45]. While pypepa is a library and a toolset for PEPA, it is not yet fully PEPA compatible. All the necessary features to make it fully PEPA compatible, plus more, are to be added in future versions [70]. These tools are being used in analytical modeling.

5.2 MATLAB (Matrix Laboratory)

MATLAB is a high level language and interactive environment for numerical computation, visualization and programming. The language, tools, and built-in math functions allow exploration of multiple approaches and reaching a solution faster than with spreadsheets or traditional programming languages such as C/C++ or Java [72]. It can be used for a range of applications including analytical modeling [71], signal processing and communications, image and video processing, control systems, test and measurement, computational finance and computational biology [72]. This tool can be deployed for both analytical and simulation modeling and measurements.

5.3 SevOne (Severity One)

SevOne (a short form of Severity One) is a next generation solution designed to monitor, measure, report and alert on the status of today's dynamic converged network environments, with the Big Data networks and provide actionable insight about the networks, applications and systems in focus [57].



Figure 3: SevOne Real-time Dashboard interface (Source: [66])

SevOne meets the demands of today's networks and technologies with the ability to measure, monitor and analyze Ethernet backhaul performance, 4G networks and quality of experience for VoIP. It supports performance analysis of both IPv4 and IPv6 [57]. Though it is a commercial software there is a free version available for download and use at www.sevone.com.

5.4 NS-3 (Network Simulator-3)

NS-3 is a discrete-event network simulator, targeted primarily for research and educational use. It is a free software and is publicly available for research, development and use. The goal of NS-3 is to develop a preferred, open simulation environment for networking research. It supports research on both IP and non-IP based networks but has been mostly used for wireless/IP simulations [58].

5.5 GNS3 (Graphical Network Simulator 3)

GNS3 is a Graphical Network Simulator that allows emulation of complex networks. It allows the emulation of Cisco IOSs on Windows, Linux and Mac OSX based computer. GNS3 uses the Scalable Vector Graphics (SVG) technology to provide high quality symbols for designing any network topology. It has been developed in python as an open source free program. The accuracy of GNS3 simulation is very high. It is a tool meant for learning and testing in a lab environment. It can also be used to experiment features or to check configuration that need to be deployed later on real devices [59]. It can be an invaluable tool for preparing for Cisco and Juniper certifications such as CCNA, CCNP and CCIE.

5.6 VNS (Vehicular Network Simulator)

VNS is a high-performance simulator for vehicular networks. VNS is a simulation framework that completely integrates the mobility and network components in a transparent and efficient way, reducing the overhead of communication and synchronization between different simulators. It provides bi-directional interaction between a microscopic mobility model and network simulators such as NS-3 and OMNET++ [60]. Closely related to VNS is VEINS (Vehicles in Network Simulation) which is another open source Inter-Vehicular Communication (IVC) simulation framework composed of an event-based network simulator and a road traffic micro simulation model. Veins is made up of two distinct simulators, OMNET++ for network simulation and SUMO (Simulation of Urban Mobility) for road traffic simulation [61]. Another one is TraNS (Traffic and Network Simulation

Environment) which is a GUI tool that integrates traffic and network simulators (SUMO and ns2) to generate realistic simulations of Vehicular Ad hoc NETWORKS (VANETs) [62]

5.7 WinEyeQ

WinEyeQ is one of the IP network performance analysis tools developed by Touchstone Technologies with particular focus on Voice over IP networks. It has support for a range of protocols such as SIP (Session Initiation Protocol), H323, MGCP (Media Gateway Control Protocol), Megaco (Media Gateway Controller), HTTP (Hypertext Transfer Protocol), SMTP (Simple Mail Transfer Protocol), POP3 (Post Office Protocol 3), FTP (File Transfer Protocol), real-time streaming protocol, SNMP (Simple Network Management Protocol) and Telnet. WinEyeQ also can capture video as well as RTP streams [63]. In addition to monitoring, measuring and reporting, it also helps diagnose where problems occur in the network. It collects the relevant information and has the ability to integrate that data into back office management systems, allowing them to troubleshoot remotely [64].



Figure 4: Wineyeq user interface (Source : [67])

5.8 Riverbed OPNET Modeler Suite

Riverbed OPNET Modeler Suite comprises a suite of protocols and technologies, and features a sophisticated development environment. It allows realistic analysis of simulated networks to compare the impact of different technology designs on end-to-end behavior through the modeling of all network types and technologies [including VoIP (Voice over Internet Protocol), TCP (Transmission Control Protocol), OSPFv3 (Open Shortest Path First version 3), MPLS (Multi-Protocol Label Switching), IPv6 (Internet Protocol version 6) and more] [65].

6. THE FUTURE FOCUS OF PERFORMANCE ANALYSIS OF NETWORKS

Performance and cost are the two primary characteristics for comparing different approaches to solving a computing or network design problem. Both characteristics can be considered along many dimensions, such as computation throughput and communication latency (performance) and power draw and price (cost). The duo possesses cross-cutting characteristics that apply to all research directions in the field of computing generally [68].

The future focus of performance analysis in computing generally will highly tilt towards prediction from performance



analysis results. This will be expected in the areas of high performance computing, high speed networks (converged and optical networks), future heterogeneous wireless networks which includes Universal Mobile Telecommunications System (UMTS), Global System for Mobile Communications (GSM), Wi-Fi (Wireless Fidelity – a trademark name used to describe all network components based on IEEE 802.11 standards), WiMAX (Worldwide Interoperability for Microwave Access) , wireless sensor networks, 4G wireless networks and Ad-hoc networks like Mobile Ad-hoc Network (MANET) and Vehicular Ad-hoc Network (VANET) [69]. The research focus in these areas will also encompass the emergent cloud technologies, software-defined networks and the next-generation protocol (IPv6).

The future roles of the academia in these areas will be all-encompassing. The academia will be expected to champion novel researches in these areas with a high level of cooperation with the technology industry.

7. CONCLUSION

In this paper, the concept of performance analysis in general is carefully considered, while focusing on computer networks in particular. The three broad techniques currently being employed in performance analysis of computer networks which are analytical modeling, simulation and measurements are explored. The nature and the requisite skills for a performance analyst were enumerated. In addition, the performance metrics of computer networks were reviewed with available software tools for carrying out network performance analysis. Finally, the new innovations and future focus of performance analysis with emphasis on computer networks were discussed. In retrospect, the field of performance analysis of computer networks is highly dynamic and progressive. It is the authors' belief that this write-up will serve as a quick inductive retinue for new researchers in the field.

8. REFERENCES

- [1] http://www.josseybass.com/legacy/rossett/rossett/what_is_pa.htm retrieved on 16th January, 2013
- [2] <http://cs.brown.edu/~rt/gdhandbook/chapters/networks.pdf> retrieved on 16th January, 2013
- [3] <http://www.ece.drexel.edu/sethu/ECE-C632/> retrieved on 16th January, 2013
- [4] Jain R., Art of Computer Systems Performance Analysis Techniques for Experimental Design Measurements Simulation and Modeling. John Wiley & Sons, 1991, pp. 16 -25
- [5] https://en.wikipedia.org/wiki/Computer_network retrieved on 16th January, 2013
- [6] <http://www.cs.purdue.edu/homes/park/cs422-intro-2-05s.pdf> retrieved on 16th January, 2013
- [7] http://staffweb.cms.gre.ac.uk/~pa10/ResMeth/network_performance_modelling.pdf retrieved on 16th January, 2013
- [8] Deru M. and Torcellini P. (2005), Performance Metrics Research Project – Final Report, A Technical Report , retrieved from <http://www.energyiq.lbl.gov> , accessed January 16, 2013.
- [9] Andreas Hanemann, Athanassios Liakopoulos, Maurizio Molina, D. Martin Swany, (2006) "A study on network performance metrics and their composition", Campus-Wide Information Systems, Vol. 23 Iss: 4, pp.268 - 282
- [10] Modeling & Performance Evaluation of Networks and Computer Systems, CS553 Course Note Slide, <http://web.cs.wpi.edu/~claypool/courses/533-S04/> retrieved on 16th January, 2013
- [11] http://www.techsoupforlibraries.org/Network_Performance_Metrics_Defined_TechSoup_for_Libraries.htm retrieved on 16th January, 2013
- [12] http://en.wikipedia.org/wiki/Network_Delay retrieved on 16th January, 2013
- [13] <http://en.wikipedia.org/wiki/Jitter> retrieved on 16th January, 2013
- [14] http://en.wikipedia.org/wiki/Digital_bandwidth_Capacity retrieved on 16th January, 2013
- [15] http://www.telchemy.com/appnotes/Telchemy_Impact_of_Delay.pdf retrieved on 16th January, 2013
- [16] Network Performance, <http://www.cs.purdue.edu/homes/park/cs422-intro-2-05s.pdf> retrieved on 16th January, 2013
- [17] Performance Metrics, <http://surendar.chandrabrown.org/teach/spr04/cse598N/Lectures/Lecture3.pdf> retrieved on 16th January, 2013
- [18] Ramon Puigjaner, Performance Modeling of Computer Networks, IFIP/ACM Latin America Networking Conference, October 3 - 5, 2003, La Paz, Bolivia, http://staffweb.cms.gre.ac.uk/~pa10/ResMeth/network_performance_modelling.pdf retrieved on 16th January, 2013
- [19] Alberto Leon-Garcia, Probability, Statistics, and Random Processes for Electrical Engineering 3rd Edition, Pearson Education, Inc. (2008), pp. 17-23
- [20] <http://mathworld.wolfram.com/MarkovChain.html> retrieved on 15/05/2013
- [21] https://en.wikipedia.org/wiki/Queueing_theory retrieved on 15/05/2013
- [22] <http://www.sti.uniurb.it/events/sfm07pe/slides/Balsamo.pdf> retrieved on 15/05/2013
- [23] Jane Hillston (2012), Queueing Networks Lecture notes retrieved from <http://www.inf.ed.ac.uk/teaching/courses/pm/Note5.pdf> on 15/05/2013
- [24] Kishor S. Trivedi, Hairong Sun (1998), Stochastic Petri Nets and Their Applications to Performance Analysis of Computer Networks, Proceedings of the Int'l Conference



- on Operational Research, retrieved from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.44.4578> on 17/5/2013
- [25] [http://camars.kaist.ac.kr/~maeng/cs710/esd07/PetriNetsProperties, Analysis and Applications.pdf](http://camars.kaist.ac.kr/~maeng/cs710/esd07/PetriNetsProperties,AnalysisandApplications.pdf) retrieved on 17/5/2013
- [26] <http://embedded.eecs.berkeley.edu/Research/hsc/class.F03/ee249/discussionpapers/PetriNets.pdf> retrieved on 17/05/2013
- [27] M. Ajmone Marsan, G. Balbo, and G. Conte, "A Class of Generalized Stochastic Petri Nets for the Performance Evaluation of Multiprocessor Systems", retrieved from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.90.8002> on 17/5/2013
- [28] Gianfrance Ciardo (1989), Analysis of Large Stochastic Petri Net Models, PHD Thesis, Department of Computer Science, Duke University retrieved from <http://www.cs.ucr.edu/~ciardo/pubs/CiardoPhdThesis.pdf> on 17/5/2013
- [29] G. Ciardo, J. Muppala, K.S. Trivedi (1992), Analyzing Concurrent and Fault-tolerant Software Using Stochastic Reward Nets, retrieved from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.16.1072> on 17/5/2013
- [30] Andrea Bobbio, Antonio Puliafito, Miklós Telek, Kishor S. Trivedi (1998), Recent Developments in Non-Markovian Stochastic Petri Nets, retrieved from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.26.5807> on 17/5/2013
- [31] Brigitte Plateau and W. J. Stewart (1997), Stochastic Automata Networks, retrieved from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.47.7863> on 17/5/2013
- [32] Anne Benoit, Brigitte Plateau and William J. Stewart (2001), Memory Efficient Iterative Methods for Stochastic Automata Networks, retrieved from <http://hal.inria.fr/inria-00072328/en/> on 17/5/2013
- [33] J. B. Othman, L. Mokdad, M. O. Cheikh and M. Sene (2009), Performance analysis of composite web services using Stochastic Automata Networks over IP network, Proceedings of the 14th IEEE Symposium on Computers and Communications (ISCC 2009), July 5-8, Sousse, Tunisia. pp. 92-97
- [34] J. Brigitte Plateau and Karim Atif (1991), *Stochastic Automata Network For Modeling Parallel Systems*, retrieved from reproject.cwi.nl/reo/export/2469/docs/QIAintoMC/related/e1093.pdf on 17/5/2013
- [35] J. Hillston (2005), *Fluid Flow Approximation of PEPA models*, Proceedings of the 2nd Int'l Conference on the Quantitative Evaluation of Systems, IEEE Computer Society, retrieved from www.dcs.ed.ac.uk/pepa/fluidflow.pdf on 17/5/2013
- [36] <http://en.wikipedia.org/wiki/PEPA> 17/5/2013
- [37] J. Hillston (1996), *A Compositional Approach to Performance Modeling*, Cambridge University Press ISBN 0-521-57189-8, retrieved from <http://www.dcs.ed.ac.uk/pepa/book.pdf> on 17/5/2013
- [38] J. Hillston (2005), Tuning systems: From composition to performance, Computer Journal, retrieved from <http://www.dcs.ed.ac.uk/pepa/tuningsystems.pdf> on 17/5/2013
- [39] <http://homepages.inf.ed.ac.uk/jeh/venice2013/Note2.pdf> retrieved on 18/5/2013
- [40] <http://www.inf.ed.ac.uk/teaching/courses/pm/Note13.pdf> retrieved on 18/5/2013
- [41] http://www.inf.utfsm.cl/~hallende/download/Simul-2-2002/Introduction_to_Modeling_and_Simulation.pdf retrieved on 18/5/2013
- [42] Harry Perros (2009), Computer Simulation Techniques: The definitive introduction!, retrieved from <http://www.csc.ncsu.edu/faculty/perros/simulation.pdf> on 18/5/2013
- [43] G. Martinovic, B. Petrisevac and D. Zagar (2010), Monitoring and measurement of computer network performance, Technical Gazette 17, 3, pp. 317-326
- [44] Network Performance Measurement and Analysis, from <http://www.cs.wisc.edu/~pb/640/perform.ppt> retrieved on 18/5/2013
- [45] Network Performance Measurement and Analysis Part 1: A Server-Based Measurement Infrastructure, <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.47.2703> retrieved on 30/5/2013
- [46] Jing Xie (2011), A Temporal Network Calculus for Performance Analysis of Computer Networks, PhD Thesis, Norwegian University of Science & Technology, Faculty of Information Technology, Mathematics & Electrical Engineering, Department of Telematics. Pp. 3-5
- [47] Joze Mohorko, Frasn Matjaz and Klampfer Sasa (2008), Advanced Modeling and Simulation Methods for Communication Networks, Microwave Review, retrieved from
- [48] List of Computer Simulation Software, http://en.wikipedia.org/wiki/List_of_computer_simulation_software retrieved 10/06/2013
- [49] Wehrle K., Günes M. and Gross J. (2010), Modeling and Tools for Network Simulation, published by Springer, ISBN 978-3-642-12331-3
- [50] CAIDA Measurement Tool Taxonomy, retrieved from <http://www.caida.org/Tools/taxonomy.htm> on 10/06/2013
- [51] CAIDA Measurement and Analysis Tools, retrieved from [http://www.caida.org/Tools/Measurement and Analysis Tools.htm](http://www.caida.org/Tools/Measurement_and_Analysis_Tools.htm) on 10/06/2013
- [52] A Survey of Network Traffic Monitoring and Analysis Tools, retrieved from on 10/06/2013
- [53] <http://en.wikipedia.org/wiki/Innovation>, retrieved on 11/06/2013
- [54] S. Gilmore and J. Hillston (1994), The PEPA Workbench: A Tool to Support a Process Algebra-based Approach to Performance Modeling,



- <http://www.dcs.ed.ac.uk/pepa/workbench.pdf> retrieved on 17/06/2013
- [55] J. Hillston and S. Gilmore (2010), PEPA tools, <http://www.dcs.ed.ac.uk/pepa/tools/> accessed on 17/06/2013
- [56] H. Akhtar (2009), An Overview of Some Network Modeling, Simulation & Performance Analysis Tools, http://www.bnnsolutions.com/OverOfSome_1.pdf retrieved on 17/06/2013
- [57] <http://www.sevone.com> accessed on 11/06/2013
- [58] <http://www.nsnam.org/overview/what-is-ns-3/> accessed on 29/08/2013
- [59] <http://www.gns3.net/documentation/gns3/introduction-to-gns3/> accessed on 29/08/2013
- [60] <http://www.dcc.fc.up.pt/~rjf/vns/> accessed on 29/08/2013
- [61] <http://veins.car2x.org> accessed on 20/09/2013
- [62] <http://lca.epfi.ch/projects/trans/> accessed on 20/09/2013
- [63] <http://www.networkworld.com/reviews/2006/091806-voip-test-analysis-touchstone.html> accessed on 29/08/2013
- [64] <http://www.tmcnet.com/channels/voip-test-solutions/articles/53944-touchstone-adds-voip-reporting-capabilities-wineyeq.htm> accessed on 29/08/2013
- [65] <http://www.riverbed.com/products-solutions/products/network-planning-simulation/Network-Simulation.html> accessed on 20/09/2013
- [66] <http://www.sevone.com/download> accessed 27/09/2013
- [67] <http://www.nihoncornet.co.jp/products/wineyeq.html> accessed 27/09/2013
- [68] <http://ft.ornl.gov/research/performance> accessed 27/09/2013
- [69] <https://www.futurenetworksummit.eu/2013/default.asp?page=c4p> accessed on 27/09/2013
- [70] <https://pypepa.readthedocs.org/en/latest/> accessed on 30/09/2013
- [71] Mom, J.M. and C.I. Ani. 2012. “An Integrated Block-Oriented Simulation Model for Estimating Cell Loss Rate in ATM Networks”, *Pacific Journal of Science and Technology*. 13(1):287-291.
- [72] <http://www.mathworks.com/products/matlab/> accessed 30/09/2013