

Simulation and Modelling of Computer Networks

Antoni Izworski¹, Slawomir Skowronski¹, and Jozef B. Lewoc²

¹ Wroclaw University of Technology, Wyb. Wyspianskiego 27, 50-370 Wroclaw, Poland
izworski@ict.pwr.wroc.pl, skowr@pwr.wroc.pl

² BPBiT Leader (Leading Designer), ul. Powst. Sl. 193/28, 53-138 Wroclaw, Poland
leader@provider.pl; <http://www.leader.wroc.pl>

Abstract. Computer networks, especially Internet, are complex systems of enormous social impacts. Numerous implementation deficiencies or even failures have been implied by inadequate simulation and analytical modelling applied for performance investigations for such systems. The paper presents some specific work in this domain on the background of a survey of the methods applied currently. The problems of robustness investigation and optimisation is also mentioned. The work presented may be summarised by the slogan: “Think computers, think performance” that has made it possible to lead successively many pioneering Polish large-scale systems and networks without any troubles connected with inadequate system performance, though the available hardware was nearly always obsolete.

1 Introduction

The present day computer networks, especially Internet, are rather complex systems needing investigations with adequate simulation and analytical modelling methods. However, the problems are frequently made light of by actual network designers and operators. An actual example [1] of a professor advising substitution of simulation investigation by a modern American computer is a typical one.

This paper tries to find an answer why it is so. Section 2 describes basic discrete-event simulation methods and tools available for investigations of computer networks and some work done specifically in the domain to carry on simulation investigations of actual real-time distributed systems and networks in an economic and efficient way. Section 3 depicts typical external network measuring tools and the internal tool designed and developed for the purposes of actual wide-area network design and operating and adapted to the needs of actual Internet-type networks. Section 4 summarizes the inadequacy of the major analytical performance methods known in the art and aimed, apparently, at computer networks and depicts, briefly, an approximate analytical model developed for actual computer networks. Section 5 describes an approach to modelling of computer network robustness and possible application of the method to actual design work, as providing some objective reasons for selection of the distributed system/network topology. The paper is completed with final remarks.

2 Simulation Methods for Performance Evaluation of Computer Systems and Networks

2.1 General

The simulation investigations have grown very rapidly in the world. For the interesting class called discrete event system simulation, numerous references are available (e.g. [2–9]). The AltaVista browser operated with the keywords “+computer +simulation +discrete +event + system” indicates roughly 150,000 positive search results.

There are available the programming languages oriented towards the simulation problems, such as SIMULA, SCL, MILITRAN, SIMSCRIPT, ASPOL, ECSS II, OSSL, COMSL, VRML, VRBL, CSL Sim, SLAM II, GPSS/H, SDX, Ptolemy, DSDS+, MODSIM, Simple_1, Pasion, WinSAAM, MathCOre, Ps-i. et al.. However, the advantages of the tools are offset by their computational efficiency which is usually too low for large-scale computer systems and networks. Therefore, various auxiliary simulation tools (helping in development of specific simulators) or simulators for some predefined object classes are proposed [6,7], covering such simulators and tools as: NETSIM (MIT), NIST (National Institute of Standards and Technology for ATM networks), INSANE (University of California, for IP over ATM), NETSIM (University of Richmond, for LANs) and less efficient, “general-purpose” simulators such as: CPSIM (Boyacon Tech. Inc.), NEST (Columbia University), REAL (Cornell University), NS (Lawrence Berkeley National Laboratory), OPNET (MIL 3), SimuNet (commercial package), Prophecy (Abstraction Software), Ipv6 with JAVA (University of Texas), QoS – Java (University of Ohio), HYSYS (Hyprotech Ltd.).

As it follows from the above survey of the simulation applications for network performance evaluation, non-trivial tasks require an individual approach. Specialised high level programming languages are hardly applicable. It is also rather infeasible to employ specialised simulators, since novel networks are, usually, not covered by them. Some specific simulation investigations carried out for more than 30 years and their basic useful results are summarised below.

2.2 The First Problem under Investigation

The first simulation investigation problem was connected with the pioneering Polish computer control system/network for a steel-rod mill. The objective of the system was to transfer fast the technological information to individual processing stations, in order to shorten the inter-lot delays (important for on-demand production). In addition, production management was to be improved due to automatic data acquisition and logging and due to emergency production control.

It was of major importance to know, that no billet could not be “lost” by the computer. In addition, it was very interesting to know the mean CPU occupancy time in order that the possible functionality scope could be assessed.

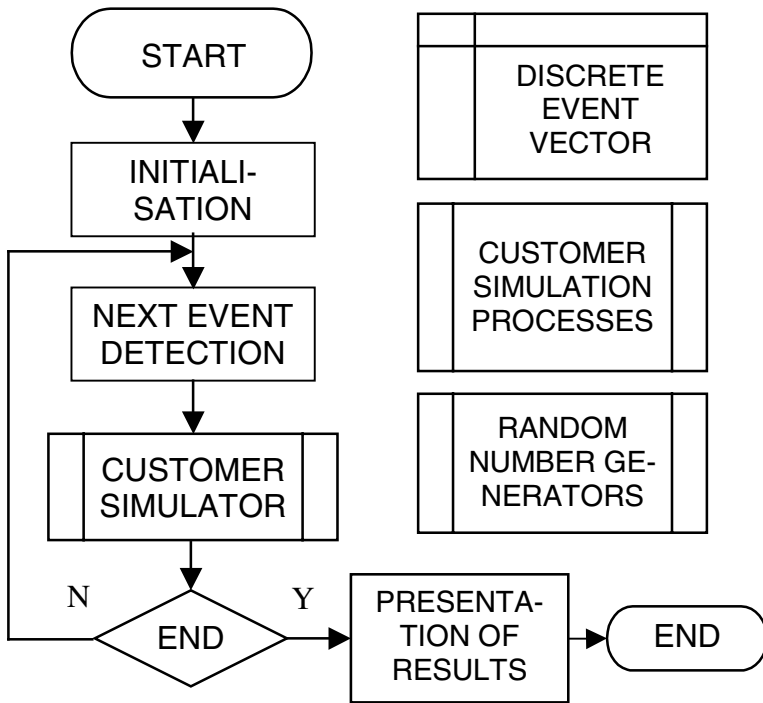


Fig. 1. Flow diagram of the discrete event simulator Des

Popular queuing theory methods of that time (pre-Leonard Kleinrock era) failed (open models were investigated and the system was closed; the input streams were not of the Poisson character and even some of them were mutually dependent; the designers were interested in intervals within which the unknown random variables were concentrated) – there were no queuing theory methods answering such questions) – then and, in most cases, even now. This was the basic reason of developing a specific Discrete event simulator (Des - see Fig. 1).

The very base of any discrete event simulator is the random number generator for the uniform distribution. From the very beginning, the author used the very fast algorithm (1) devised “by mistake” by Wrocław mathematicians.

$$\begin{aligned}
 x &:= x0 + 2 \cdot x1 \pmod{1} \\
 x0 &:= x1 \\
 x1 &:= x
 \end{aligned}
 \tag{1}$$

The figure 2, inserted by an operator, ensured rather low correlation between successive realisations of the random number, while maintaining the high speed (the multiplication needed a single shift only). This speed was also increased through application of the interval-based approximation of the distribution density for other distributions (in spite of “simulation” purists requiring some distribution tests rather inappropriate for approximation errors of several percent). Another solution increasing the Des efficiency was that of the table-looking procedure (based on so called “divide-by-two” principle, which is theoretically optimum one).

2.3 Exemplary Systems Investigated by Simulation and Results Obtained

The Rod Mill Control and Co-ordination System. In the case described above, as usual, one of the most important results for the system designer is the detailed knowledge of what actually happens in his or her system. The design is, most commonly, developed on the basis of the system functionality and, normally, there is no time to deal with detailed performance problems. The understanding of his or her solutions by the designer often enables e.g. to assess rather accurately the unknown variable distribution concentration intervals. Such simulation “by-products” were very important in each investigation case. They are the basis of the design philosophy defined when developing the exemplary computer systems/networks: “Think computers, thing performance”, that was the basic reason for that there were never problems connected with performance of the systems/ networks designed. And it is worthwhile to mention that, usually, the hardware available for the systems/networks of 5-10 years behind that applied in the developed countries.

For the specific case, the basic findings from the simulation were as follows:

- the computing power was used at a minimum extent,
- the hard real-time requirements were met by the system,
- no virtual (simulated) interrupt signal was found,
- the maximum queue lengths did not endangered with system congestion,
- the hardware and software structures designed made the system feasible.

The Radio-Astronomy Antenna Control System. The next project led was the initial design (extended feasibility study) of the pioneering Polish computer control system planned for a radio-astronomy centre. Due to the huge amount data to be recorded and analysed, limited capabilities of the available computer and a very specific control peripherals designed, the system needed very careful performance analysis and optimisation. The simulation model Des described above was used for this purposes.

The results obtained enabled to state that the hardware and software solutions proposed could fulfil the needed function in real-time, to optimise the priority assignments and to warn of high maximum service time differences (reaching even an order of magnitude) at the similar mean values, needing special caution when interpreting the mean values possibly available from the queuing theory.

System SAPI ODM. The next project was the pioneering Polish power distribution monitoring and control system SAPI ODM. Considering the fact that the power industry requirements are, usually, much more severe than those of other industries, and that the computer available was originally intended for numerical computations only, and needed specific peripherals, interfaces and software, performance evaluation tasks were of the highest importance.

The Des simulator described for the rod-mill above was adapted and used in the investigations. First of all, it was verified that the system was able to acquire and process many thousand analogue and digital signals at a high signal change rate in emergency states (this signal change rate was even much higher for the version intended for big power plant/power generating unit monitors). It was also possible to

optimise operation of the communication peripherals/networks (visual displays, teletypes, printers, telemechanique stations). For another application of SAPI ODM, in e-Training, developed successively on the computer of the computing power of some order of magnitude lower than that commonly considered as the minimum necessary for power-industry training simulators, careful simulation studies enabled to define the range of possible services and to prepare adequate training methods (pre-defined responses instead of solving complex differential equations).

All systems described above proved to be very useful and profitable (benefits gained: ca US\$ 40 billion during some 15 or so years of operation; more than 90% of Polish power system control engineers trained successively on the power system simulators).

Interuniversity Computer Network. The Interuniversity Computer Network (MSK) project was the Polish pioneering solution in networking (in wide-area networks of the public character). This was a research undertaking aimed at recognition of the phenomena in computer networks and at education of local networking staff.

An authoring solution was developed in order to investigate analytically the performance of computer networks. The Des simulator described above was adapted for the purposes and used in tuning and validation of the analytical approximations. The utilitarian, research and educational goals of MSK were achieved successfully.

3 Internal Measuring Tools

For computer network performance measurements and testing, special-purpose performance measuring tools are developed. They may be located outside or inside the network (outside and internal tools, respectively). Internal tools can be hardly found in references because of the morbid confidentiality policy of big PTO's (Public Telephone Operators); most descriptions available in public [10] concern the external ones. The latter, however, does not enable to take overall network measurements needed to verify and optimise the basic network solutions.

For validation and optimisation of MSK, an internal tool (Sitwa from Polish "System intensywnego testowania wezla") was developed and, presently, adapted to current networks operating under the TCP/IP protocol suite [1], [12], [13] (see Fig. 2).

With Sitwa, the simplifying assumptions adopted in Des (first of all, omission of flow control packets and frames in simulation) were validated positively. In addition, several dozen measuring cycles were performed, enabling performance evaluation of existing and planned network configuration (Sitwa makes it possible to measure virtual network and subscriber equipment) and validation of the Closed-route-based approximate (performance evaluation) method (Cram) devised. An interesting finding of Cram, confirmed by measurements with Sitwa, was that the phenomenon of congestion (decreased throughput at decreased thinking times) occurs also in networks with end-to end acknowledgement without any loss of packets.

A "by-product" of Sitwa was the better understanding of the phenomena occurring within any network what led to modelling of networks as superposition of closed routes (connection, calls) and to development of Cram.

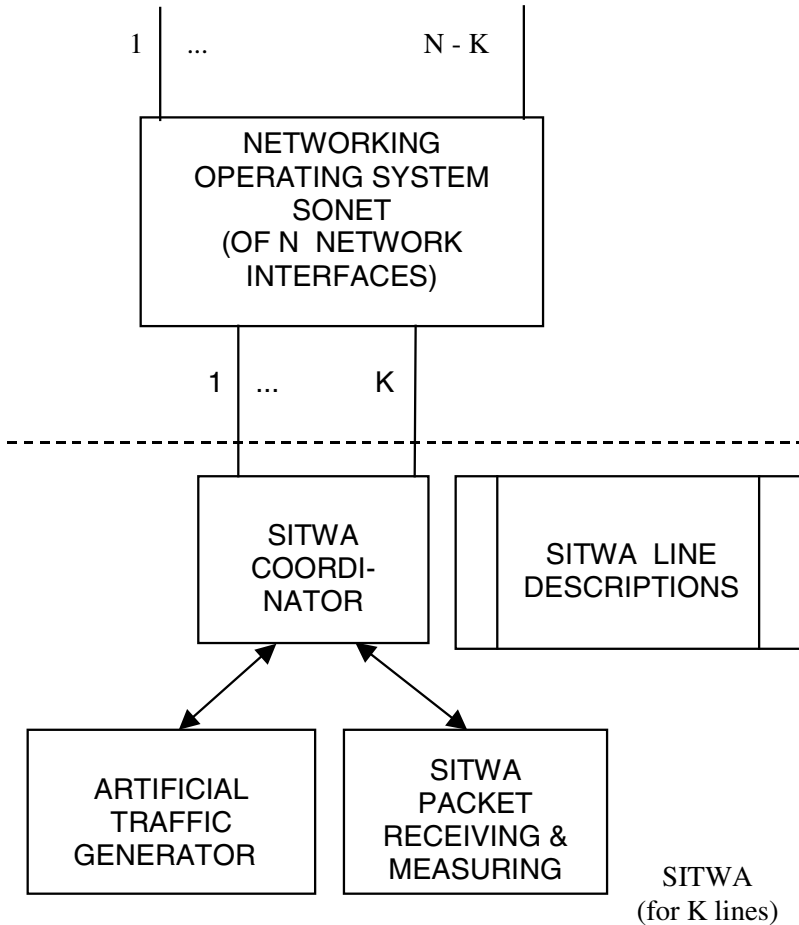


Fig. 2. Functional block diagram of Sitwa in a networking operating system Sonet

4 Analytical Performance Evaluation Methods

Instead of rather big expectancies, actual computer and communication network designers or operators can hardly use the analytical methods to evaluate (to say nothing about optimisation) the performance of networks being developed and/or operated by them. Starting from [14] to [15], the Mean Value Analysis (MVA) method has been probably the most popular method of the classic queuing network theory. However, in the method, there is employed the standard assumption: the transition probabilities for entities from line s to the next line s' are given. This is obviously false for actual computer and communication networks.

An actual packet-switching network, such as a network operating under the TC/IP protocol suite, should be analysed as a Kelly network [16], where the sojourn time at a node depends on all entities (packets) in the network. However, in spite of that the Kelly networks have been defined more than two decades ago, no classic queuing

theory methods are available for performance evaluation for this case of networks, enabling rather close approximation of actual networks.

Many stochastic queuing system performance evaluation methods have been developed during a few last dozen years [17]. They cover also the interesting case of the Kelly networks (also the more general class of the Whittle networks). However, the stochastic methods are basically theoretical mathematical tools that are not suitable for actual network designers and operators, and closed networks always produce significant problems [18].

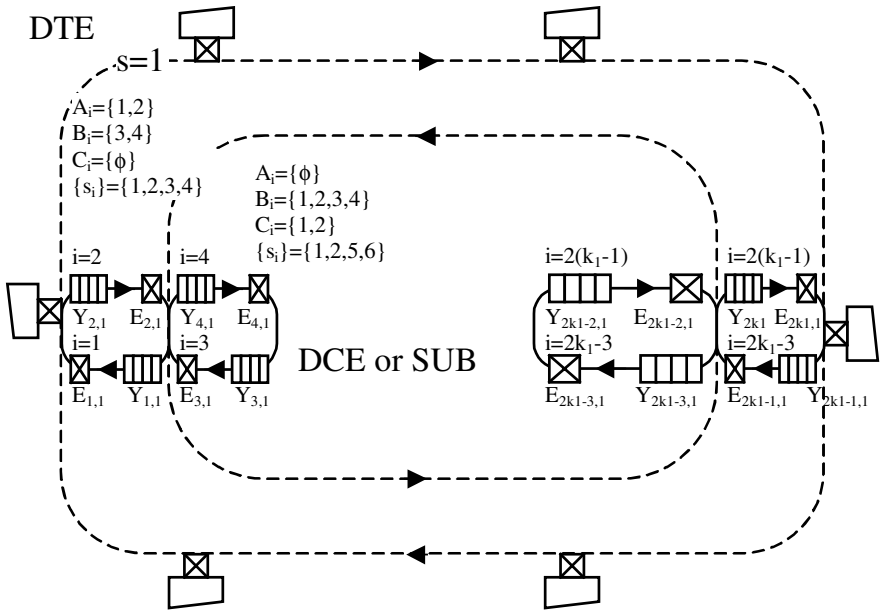


Fig. 3. Illustrative diagram of a closed-route-based network

This was the reason for development of the Cram (Closed-route-based (see Fig. 3) approximate analytical method) [1], [13]. The method was applied and validated successfully (no relative error in comparison to simulation and measurement results, exceeding 5%, was found) for various accepted and planned configurations of MSK, star-topology networks, various WANs, LANs and metropolitan networks, as well as, lately, to Computer Integrated Manufacturing and Management (CIMM) networks.

5 Network Robustness Studies

In the domain of automatic control systems, the notion of robustness, i.e. the capability of disturbance tolerance, of operation in uncertainty conditions [19]÷[24]. Already in 1982, J. Doyle [19] proposed so called function μ (structural singular value) as the system robustness measure. This function was applied to compare distributed control system topologies on some reasonable criteria (see Fig. 4). Due to the fact that distributed computer control system (networks) may be considered as

composed of time-delay type automatic control members only, the computations needed to evaluate the function μ are rather simple and the method may be applied in actual network design. An interesting finding was that the networks of the “obsolete” star topology are much more robust than the “common-medium” ones.

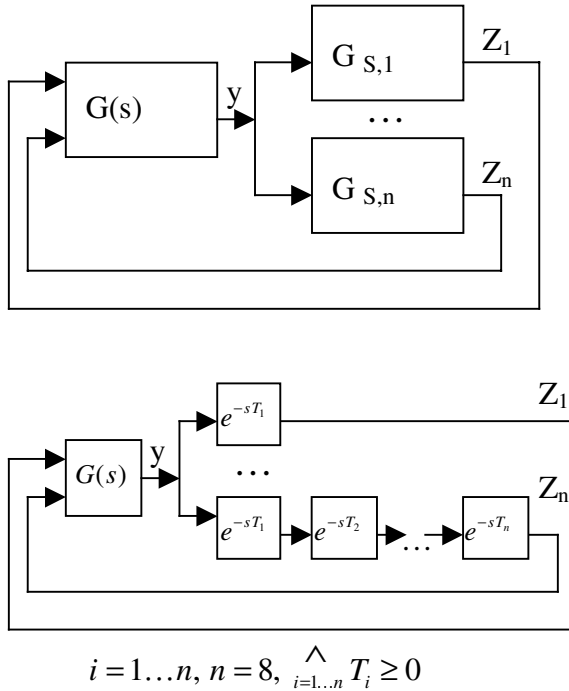


Fig. 4. Basic diagrams for the star (upper) and common-medium (lower) cases

6 Final Remarks

The simulation and analytical modelling methods known in the art are hardly applicable to actual network design and operating. The main reasons for such statement are:

- inadequate assumptions adopted in description of networks under investigations,
- theoretical models that are too complex and inapplicable to actual design,
- morbid confidentiality policy of large-scale PTOs.

The basic explanation for the successes informed of in the present paper is that the designer (the leading designer of all systems specified, except the first one for the rod-mill) was, at the same time, involved in the simulation and analytical modelling problems. This needed some wide education bases (the degree in Electronic Engineering (Computers) made up by M.Sc. with high honour in Maths). This also needed a lot of volunteer work (or the work for the Society, as it is called in Poland) since during more-than 35-year engineering career, financial resources for the

research work were available to the author only once. But, considering the technical and financial successes achieved, it was much more than worthwhile. And, additionally, the simulation and analytical modelling methods developed in the past and adapted at present may be applied for many other network design, development and operating tasks.

References

1. Izworski A. And Lewoc J.B., Approximate Analytical Performance Modelling of a Computer Integrated Manufacturing and Management System, In: Troch I. And Breitenecker F. (Editors), 4-th IMACS Symposium on Mathematical Modelling, Vienna, 2003 (accepted for publishing).
2. Kelton W., Sadowski R. and Sadowski D., Simulation with Arena, McGraw-Hill, 1998.
3. Cassandras C. and LaFortune S., Introduction to Discrete Event Systems, Kluwer, 1999.
4. Harrington J. and Tumay K., Simulation Modeling Methods: To Reduce Risks and Increase Performance, McGraw-Hill, 2000.
5. Andrea E., A Collection of Modelling and Simulation Resources on the Internet, Available at: www.idsia.ch/%7Eandrea/simtools.html, Internet, 2002.
6. Ernst T. (INRIA), Notes about network simulators, Available at: www.inrialpes.fr/planete/people/ernst, 2000.
7. Raczynski S., Simulation Encyclopedia, Available at: www.raczynski.com/pn/encyk.htm, 2002.
8. The MathWorks, Inc., Hyprotech Lifecycle Innovation, Available at www.mathworks.com/products/connections/product_main.shtml?prod_id=81, 2002.
9. Evans J. and Olson D., Introduction to Simulation and Risk Analysis, Prentice Hall, 2002.
10. Les Cottrell (Stanford University), Network Monitoring Tools, Available at: www.slac.stanford.edu/xorg/nmtf/nmtf-tools.html, 2002.
11. Izworski A., Lewoc J.B and S. Skowronski, An Internal Network Traffic Modelling/Performance Measuring Tool, In: Architectures for Quality of Service in the Internet, Warsaw, 2003 (submitted for publication).
12. Yanovitz, J. Under the Hood of Internet: An Overview of the TCP/IP Protocol Suite, Available at: <http://info.acm.org/crossroads/xrds1-1/tcpjmy.html>, 2002.
13. Izworski A., Lewoc J.B. and Skowronski S., An Approximate Actual Network Performance Evaluation Method, In: Providing QoS in Heterogenous Environmets (the 18-th International TELETRAFFIC CONGRESS), Berlin, 2003 (submitted for publication).
14. Reiser, M., Mean-value analysis and convolution method for queue dependent servers in closed queuing networks, Performance Evaluation 1, North Holland, 1981.
15. Robertazzi, T.G., Computer Network and Systems. Queuing Theory and Performance Evaluation, Springer-Verlag, 2000.
16. Kelly, F.P., Reversibility and Stochastic Networks, Wiley, 1980.
17. Surfozo, R., Introduction to Stochastic Networks, Springer-Verlag, 1999.
18. Rolski, T., Private communication, Wroclaw, 2002.
19. Doyle J., Analysis of Feedback Systems with Structured Uncertainties, IEE Proc., Vol. 129, No. 6, 1982.
20. Maciejowski J.M., Multivariable Feedback Design, Addison Wesley, 1989.
21. Hu S.S et al., Nonlinear Robust Controller Design Based on H-infinity and Mu-synthesis, IFAC Congress, Beijijn, 1999.
22. Chen J. and Gu G., Control-oriented System Identification – An H_infinity Approach, Wiley, 2000.

23. Izworski A., Lewoc J.B and S. Skowronski, Robust Performance Case Study: Topology of System Media, In: Control Application of Optimisation (IFAC), Visegrad, 2003 (submitted for publication).
24. Izworski A. and Lewoc J.B., Robustness Comparison of Enterprise Energy Distribution Systems of Various Topologies, In: Robust Control Design, Milan, 2003 (submitted for publication).