STUDIA INFORMATICA
Nr 2(13) Systems and information technology 2009

Traffic control algorithms in computer networks. Part 2 – Simulation researches of the algorithms

Andrzej Barczak, Tomasz Sitkiewicz
Institute of Computer Science, University of Podlasie
St. 3 Maja 54, 08-110 Siedlce, Poland

Abstract. This article is the continuation of the first part entitled *Review of algorithms.* In the article presented were the author's simulator of a network and the tests performed with its help. In the first part the concept and functionality of the created simulator were described. The concept emerged on the basis of an analysis of existing solutions, as well as needs the created software has to fulfill. The second part of the article depicts the testing environment, i.e. network topology and simulation parameters used during the research. What is more, in the aforementioned part, provided was the comparison of the research run with the use of the author's software and its interpretation.

Keywords. traffic control algorithms, computer networks, traffic simulation in networks.

1 The characteristics of the simulator

In order to research the traffic control algorithms in the network, effort was made to create author's simulator. In the following chapter the concept of the network simulator is described.

The developed simulator, shall be used for the research concerning selected traffic control algorithms in the computer networks. The results shall allow for experimental testing of the implemented control algorithms within specified assessment criteria. Fig. 1.1 presents a general functional diagram of the created simulator.

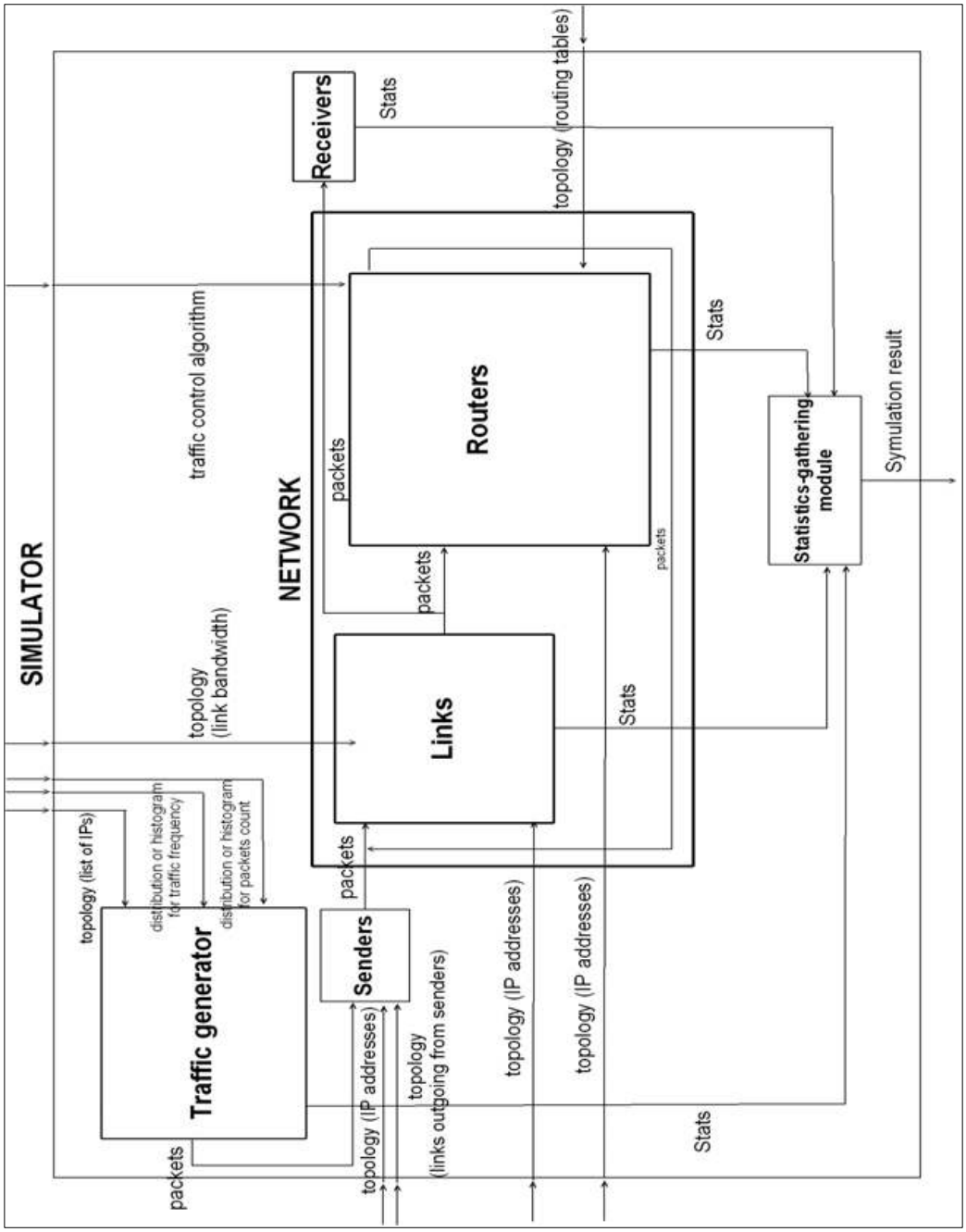

Figure 1.1 Functional diagram of the simulator

As it is presented on the Fig. 1.1, the functional model of the simulator may be divided into several main parts, like: traffic generator, senders, receivers, network and summation module. The aforementioned components are the main elements of the simulator model and are presented in detail below.

The input data of the simulator

As can be seen on the diagram in the Fig. 1.1, network typology, particularly IP addresses of the senders, receivers and routers, may be counted among the input data of the simulator. Furthermore, according to the data from the network typology, the input data of the simulator are the bandwidth of the connection line, routing tables of the routers and the traffic control algorithms in a given router. Aside from the typology the input data constitute the schedules, histograms recording the traffic frequency in the network, as well as the size of the packets generated.

Traffic generator

The functional diagram of the *Traffic generator* module, presented on the general functional diagram from the Fig. 1.1, is depicted on the Fig. 1.2.

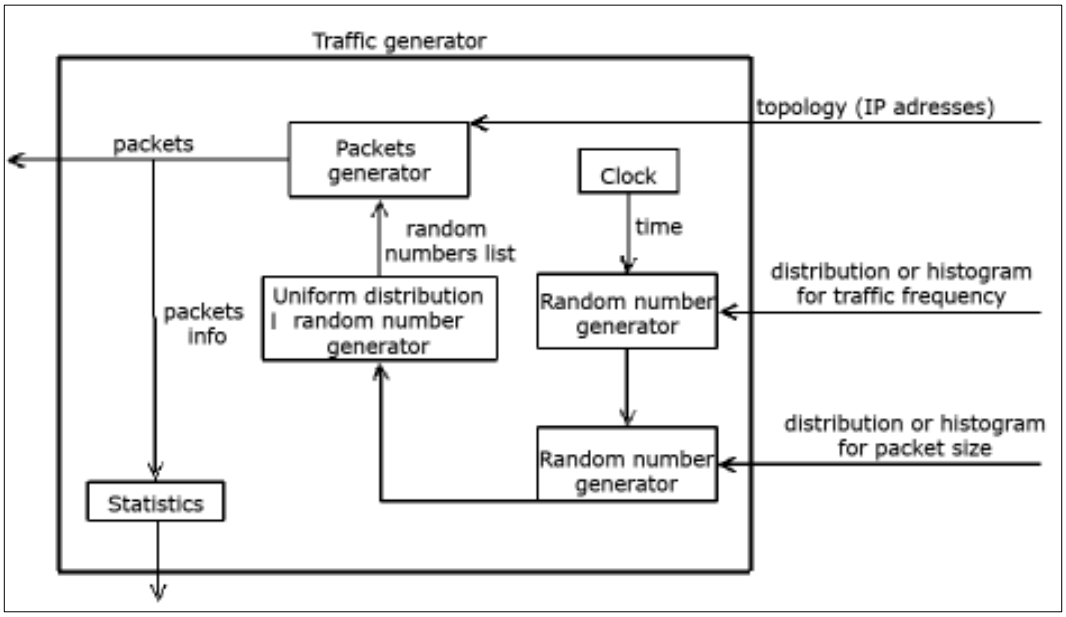

Figure 1.2 Functional diagram of the *Traffic generator* module

Traffic generator is the key element of the network simulator. This particular element creates the traffic in the network, from the intensity and time fluctuations of which depends the correctness of the simulation results obtained.

The presented generator is provided with such data as random variables distribution for the size of the packets and the IP addresses list of the senders and receivers available. As an output it returns the packets generated, which in turn reach the senders, as well as statistical information on packets, allowing for the summation module to calculate the assessment criteria.

The main component of the traffic generator (as presented on the Fig. 1.2) is the timer providing the time of the simulation. It is a discrete time, on the basis of which the real time of the simulation is calculated.

"On-off" model is a well known and commonly used solution in the process of creating the traffic generators. The generator of this type determines on the basis

of the probabilities distribution when the "on" state (when the generator sends) occurs and how long it lasts. In the "on" state the generator sends the packets to the network.

The base of the "on-off" generator functioning is the rule, that in the "on" position it generates an action (sending of a packet), whereas in the "off: state it remains idle. Such action can be presented as follows:

$$X(t) = \begin{Bmatrix} 0, \textit{when in time t is} \text{ on} \\ 1, \textit{when in time t is} \text{ off} \end{Bmatrix} \tag{1.1}$$

where:

t – time >=0,

X – a process in a time function.

For the generator to reflect the ISO/OSI model effectively, *X* function must be created for every layer of the model. Every layer works with different frequency. Approximate values of the frequencies for the given layers for [4] are shown in the Table 1.1.

Table 1.1 Layers of the ISO/OSI model and their work frequencies in the simulator

No.	Layer	Frequency
1	User	$f \leq 10^{-2}$Hz
2	Application	10^{-2}Hz$\leq f \leq$1Hz
3	Presentation	10≤f≤103Hz
4	Session	
5	Transport	
6	Network	
7	Data link	103Hz≤f

Furthermore, at the output of every layer (except for the user) must be a buffer responsible for storing the packets awaiting the access to the next, lower layer.

On the basis of such characteristics of individual layers, traffic in the physical layer may be presented with a formula:

$$X_p(t + \Delta t_{ps}) = \prod_{i=1}^{n} \left[X_i(t + \Delta t_{pi}) + X_i^b(t + \Delta t_{pi}^b) \right] \tag{1.2}$$

where:

n – number of layers,

X_i – X function for *i*th layer,

X_i^b – function for *i*th buffer,

Δt_{pi} – propagation time of the signal in a layer,

Δt_{pi}^b – propagation time of the signal in a buffer.

For every layer, the "on" and "off" phases occur in accordance with a selected probability distribution. Δt values are the time periods, for which the generator for the over-layer remains in the "off" state. The periods reflect the delays in the computer system. They are calculated on the basis of the time interval generating in keeping with the distribution for the "off" phase and multiplying by a random (of a uniform distribution) number from the (0,1) range.

The next module requiring sampling, is the module determining the sender and the receiver of the packet. This random-numbers generator should sample the numbers of a uniform distribution.

The last module of the traffic generator is a packet generator, which creates a packet on the basis of the address list of the senders and receivers, as well as formerly prescribed random values and transfers it to a proper sender, that means introducing it to the network.

Apart from the depicted functional modules there is also statistics-gathering module, i.e. the data concerning in what period of time the packet was sent from what sender to what receivers.

At this point worth mentioning are the data, which every single simulated packet should carry. Aside from the obvious value being the size of the packet, of key importance, from the simulator's point of view, are such data as IP address of the sender and the IP address of the receivers. It is also reasonable (in such a functional model) to determine the IP address of the nod, which it should reach during the next stage.

Senders

The *Senders* module presented on the functional diagram on the Fig. 1.1 is responsible for transferring of the generated packet to a proper connection. On the input it accepts the IP addresses of all the senders, list of connections, which originate from a given sender, as well as packets generated by the *Traffic generator* module. On the output it returns the packets, which were transferred by the generator, in order to send them to an appropriate connection.

Links

The module called Network consists of two basic elements: *Links* and *Routers.*

The *Links* module, as presented on a diagram on Fig. 1.1, on the input receives the data concerning the topology in the form of bandwidth and IP addresses of the nods at the beginnings and ends of the connection line. What is more, it receives the packets from the Senders module. At the output of this module the packets are returned, which are properly directed to the Routers or Receivers modules, as well as statistical data, on the basis of which, the summation module shall be able to calculate the criteria of the assessment.

A functional diagram of the *Links* module is placed on the Fig. 1.3.

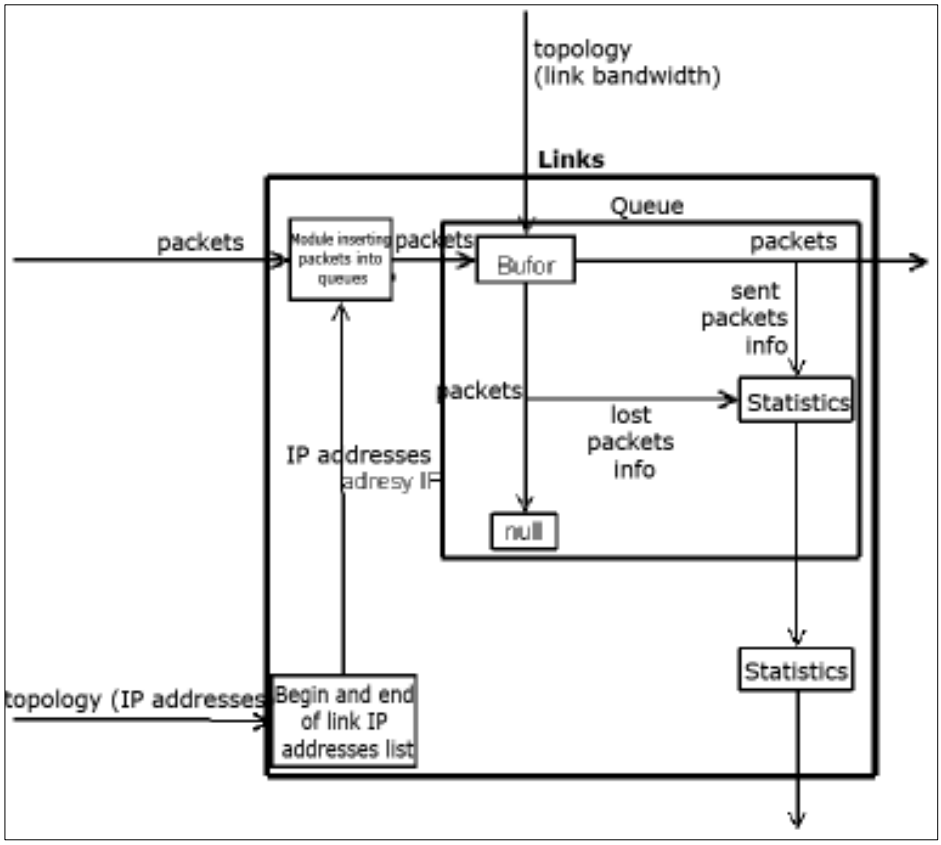

Figure 1.3 Functional diagram of the *Links* module

Links indicate, in general terms, a list of all the connections in the network. This module contains the IP list of the nods at the beginning and end of every connection line. Aside from that, the *Link* has a *queue* which allows to control the flow of the packets depending on the bandwidth. Needed is also a module, which on the basis of the addresses of the packets entering the network, assigns them to the proper connection lines and places them in the right queues. Queue components consist of: a *buffer*, null packet module, statistics gathering module. It is a FIFO (first in first out) queue of a drop tail type, that means that the first packet entering the queue will be drawn, while in the case of buffer overflow the last coming packet is lost.

From a queue of a given connection line a packet may go to (depending on what can be found at the end of a connection line) a router or a receiver. If the packet stays within the *network* module, it will be transmitted to the routers module.

Routers

The *Routers* module receives as an input information the data from the topology, i.e. IP addresses of the routers present in the network and the routing tables of those routers. Furthermore, this module receives the information concerning the traffic control algorithm which must be used by the routers and the packets directed to the routers from the *Links* module. The packets directed into the proper connec-

tion lines, as well as statistical data entering the summation module, are the output data.

Fig. 1.4. presents the functional diagram of the *Routers* module.

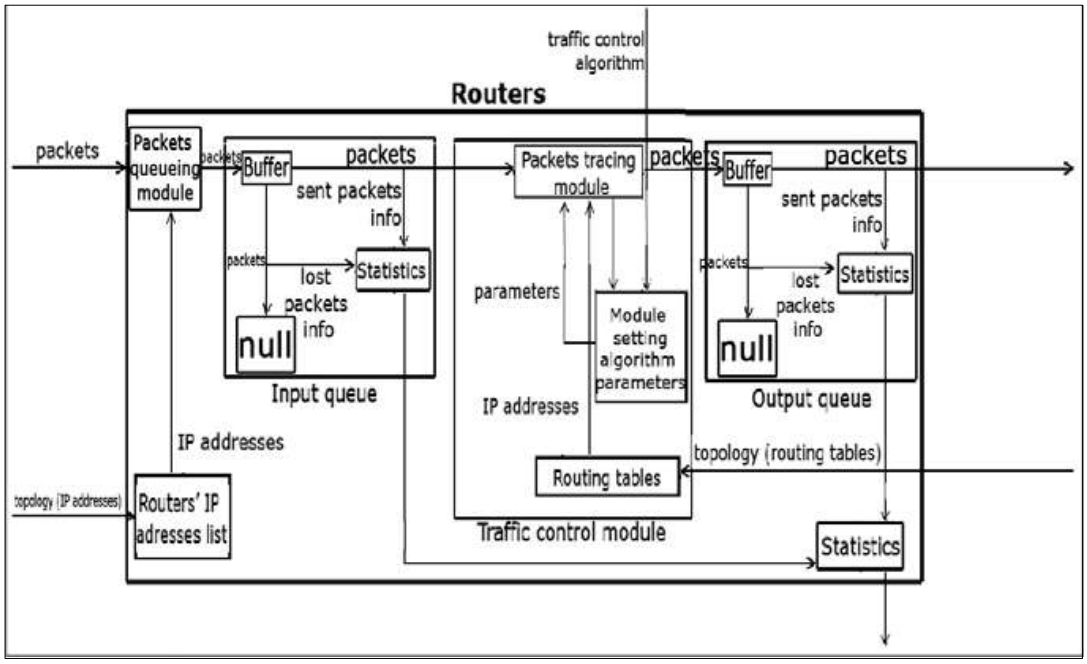

Figure 1.4 Functional diagram of the *Routers* module

The *Routers* module, just as *links,* constitutes the representation of all the routers in the network. It contains the IP addresses list of all the routers in the network, a module assigning the incoming packet to the input queue of a proper router. Furthermore, the *routers* module consists of such elements as*: input queue*, *traffic-control module, output* queue and the *statistics-gathering* module. *Queues* are the components analogical to those presented in the *links* module, with an exception being, that the buffer of those queues is not regulated by the bandwidth, but has only a fixed capacity. A packet received from an input queue gets into the traffic-control module. This component establishes the route of the packet on the basis of the routing tables and traffic-control algorithm's parameters. The module changing the parameters of the algorithm reacts to every established route of a packet and corrects the algorithm parameters accordingly. Some control algorithms take control through the parameters in the traffic sources. For those algorithms a feedback module has been created, which transmits proper values of the parameters into the traffic sources. The packet with an assigned route enters into an output queue of the router, from which it returns to the *links* module, from where it is transmitted further on (to the next routers or receivers).

Receivers

Receivers module is a so called packet sink, it receives the packets, stores the statistical data concerning their arrival, delay, source, etc. and deletes them.

Statistics-gathering module

The data from all the modules gathering the statistics get to the *statistics-gathering module.* This module gathers the data from the traffic generator, links, routers and receivers. The module provides the output data from the simulator. Its main role is the calculation to what extent the assessment criteria have been fulfilled.

Output data of the simulator

After the simulation, the application should generate the results, which in turn shall be used for research and analysis of the traffic control algorithms. To the obvious data, which the simulator should generate, one includes such parameters as duration of the simulation, the number of traffic sources, number of traffic destinations. Furthermore, the simulator should calculate the statistics concerning the fulfillment of the assessment criteria.

2 Research environment

This chapter includes the description of a demonstration network topology used in the research of algorithms and the parameters of the simulation characteristic for every simulation run in the course of the research.

2.1 Network topology

The diagram of the network, being the subject of the research, is based on the ECN (Engineering Computer Network) network on Purdue University in USA. In this network work the students, senior researchers and lab technicians of the University.

The network is composed of such university departments as: Mechanical Engineering, Chemical Engineering, Agricultural & Biological Engineering, Nuclear Engineering, Aeronautics & Astronautics, Industrial Engineering, Civil Engineering, Electrical & Computer Engineering, Materials Engineering, Engineering Computer Network.

The topology used for research is based on the real university network. However, it is to some extent a simplified model of this network (mainly because of a high number of computers).

The main router connecting the Intranet network is the two-Gigabit router. Ten one-Gigabit routers (one for every university department) are connected to it. All of the computers in a given department are connected to a proper router.

In general, in the modeled network, there are 60 computer of x86 type, 41 Macintosh computers, 10 HP, 8 SGI, 38 SUN Enterprise, 8 SUN Ultra, 12 ALPHA, 3 SUN enterprise, 11 SUN Fire, 26 x86 Server.

Fig. 2.1. Presents the diagram of a network including the speed of the connections within the subnetworks.

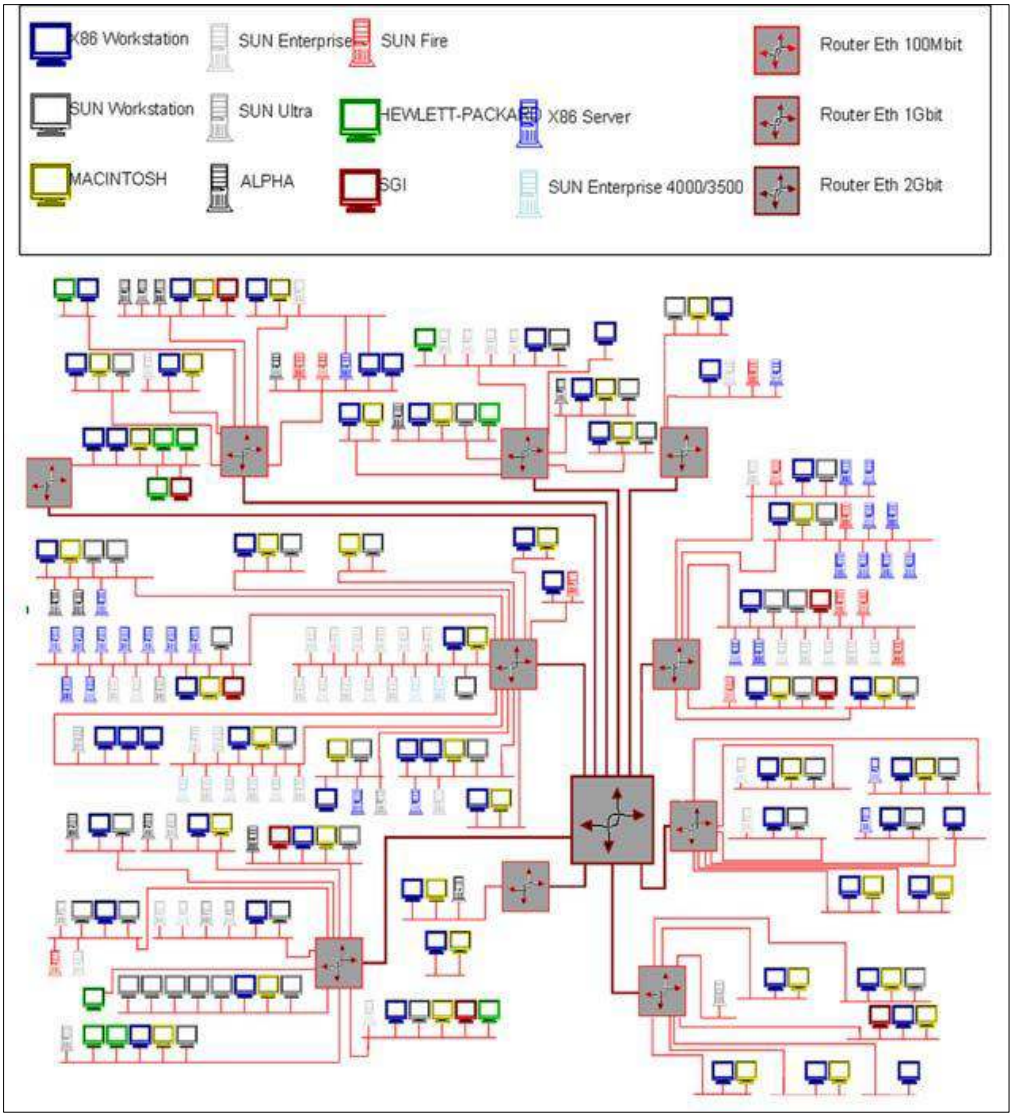

Figure 2.1 The diagram of the network topology used for research

2.2 Network traffic characteristics

On the websites of the university [2], on the network of which the described topology is based, there is an extensive data base comprising the information on the traffic in this network. On this basis it is possible to determine the traffic load of the network (calculated according to the number of data, number of requests, number of sessions, duration of a session) weekly (histograms for every day), or daily (histograms for every hour).

In order to perform the tests, real data have been used concerning the traffic in the network presented on the Fig. 3.1 from 16 – 22 February 2009. The university makes the data on the traffic available in the form of monthly, weekly, daily, hourly histograms.

The data in this form cannot be used to supply the developed simulator, that is why they had to be properly transformed. Ultimately the traffic generator in the

simulated network is supplied with packet size histograms, as well as inter-arrival time(IAT) histogram.

In order to create the packet-size histogram, the summation of the number of packets sent in every hour of the time tested (seven days) and the average packet size sent in this time, has been calculated. Next, the scope of the packet size has been divided into even partitions and the number of packets has been assigned to those partitions. Fig. 2.2 presents a histogram, created on the basis of this data.

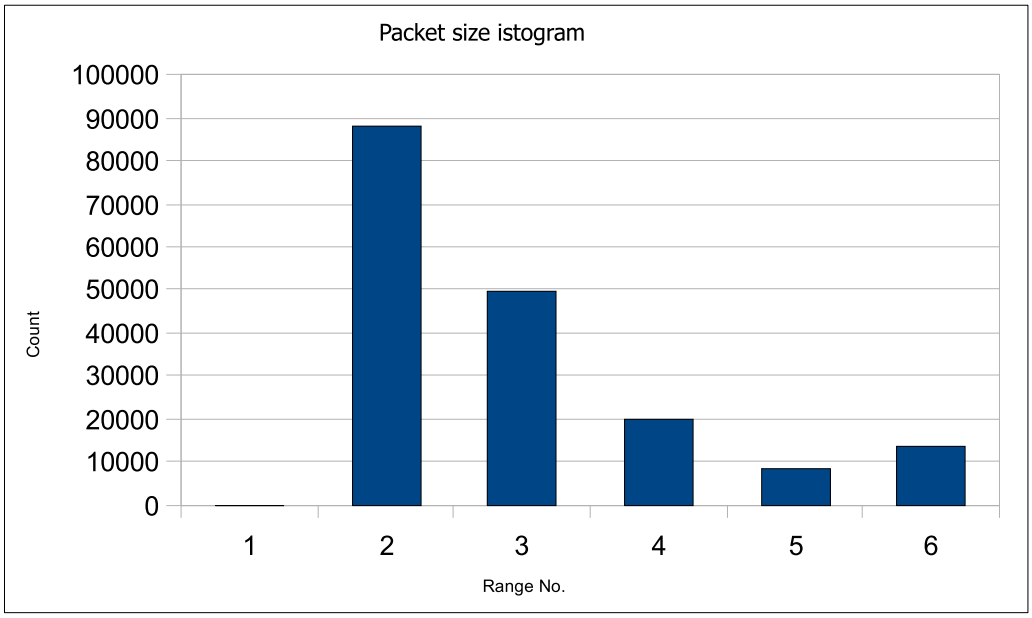

Figure 2.2 Packet-size histogram

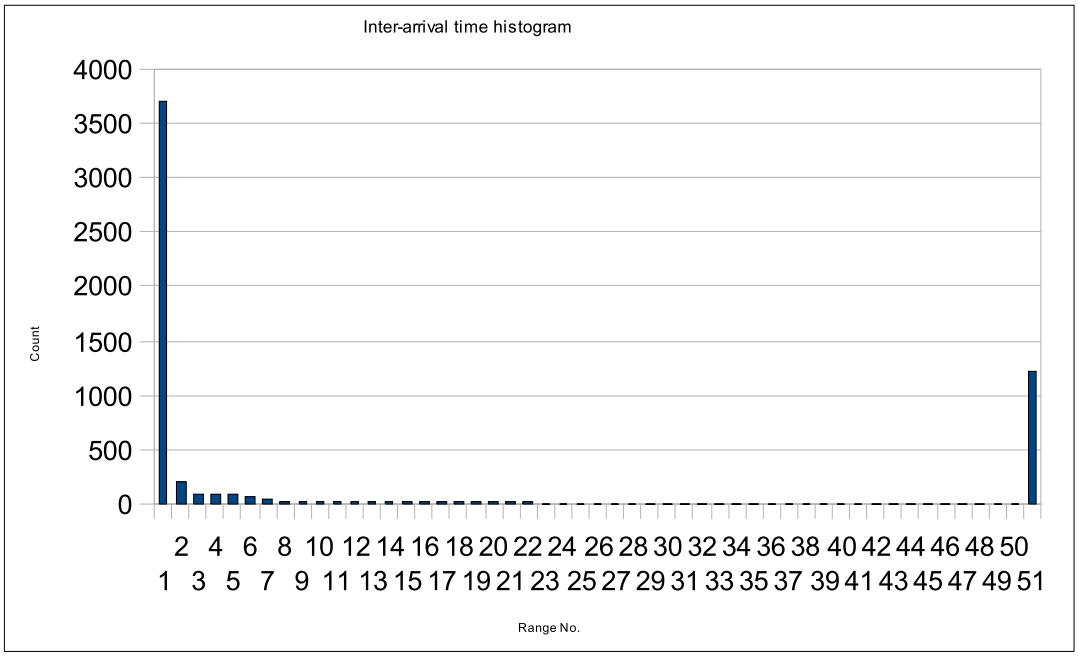

Figure 2.3 Inter-arrival time histogram

Analogical procedure has been applied to the inter-arrival time histogram. Also used, were the information available on the website of the Purdue University. The calculated duration of the inter-arrival time has been divided into 50 equal partitions. Every partition has fixed number of packets, which arrived after the time assigned for a given partition. On the basis of this data a histogram presented on the Fig. 2.3 has been developed.

2.3 Parameters of the simulation

In order to perform a comparison analysis of the implemented traffic-control algorithms in the computer network, with the help of the created simulator, a series of tests has been run.

For every traffic control algorithm (RIP, IGRP, OSPF, ISIS and EIGRP) one topology file has been prepared, as well as a reference file with a static routing, where the routs have been entered manually and they were not found or modified in the simulation run. Table 2.1 presents a volume summary of the topology used for research.

Table 2.1 Volume summation in the topology

Components	Count
sources	255
destinations	255
computers	255
routers	11
links	275
queues	572

For every algorithm, tests have been performed with the use of different feed for the traffic generator. A simulation using histogram supply and three distributions – Weitbull, Pareto and exponential, has been created. In accordance with the descriptions in the materials [7] and [4] proper distribution parameters have been selected. Their summation is presented in the Table 2.2.

Table 2.2 Distribution parameters used for research.

Distribution	Scale	Shape	Position
Weibull	50	5	-
Pareto	-	1	0,01
Exponential	0,05	-	-

For every mode of the generator supply and for every algorithm, the simulation was run three times in order to avoid the distortion of the results. In the next chapters the rough values from those runs undergo an analysis and interpretation.

The data gathered during every run of the simulation are the numerical values presenting the degree of fulfillment of individual assessment criteria of the algorithms.

2.4 Traffic control algorithm characteristics

Traffic control algorithms used in described simulator can be divided into two classes – distance – vector algorithms and link state algorithms. The first class contains such protocols as RIP or IGRP. This protocols define direction and number of jumps for every link in the network. Every node in network propagate its routing table periodically to all neighbours. This propagation takes place even when no changes in routing tables are made.

In the second class can be placed OSPF and IS-IS algorithm. These algorithms also propagate information about network state. In case of change in network, the router that spot change sends link state message to its neighbours.

The RIP is the distance-vector protocol. It stores information about direction and distance to every node in network. If there are many paths to specific node, it chooses the shortest one (with lowest number of hops). The major disadvantage of this attitude is that the shortest distance is meant as the lowest number of hops which not always means the fastest connection between source and destination. Another disadvantage is that highest numer of hops is set to 15, which makes scaling of the network very difficult.

The IGRP is also distance – vector protocol. It was introduced by Cisco Systems. It is described by five characteristics – delay, bandwidth, load and reliability. By default it uses only delay and bandwidth. The distance between nodes are calculating using these characteristics. Routing informations are propagated periodically and when change takes place. Top limit of hop numer is 255, which eliminates main disadvantage of RIP.

The EIGRP is also Cisco Systems' algorithm, meant to be successor of the IGRP. It mainly works using distance-vector but also have some characteristics native for link state protocols (ie. maintaining relations with neighbourrs and routing topology). Because of this it is called hybrid protocol.

The OSPF is link state protocol. It was created to eliminate scaling problems of RIP protocol. OSPF works inside some area, which size is limited to 500 routers. The network topology is known for every router. Informations about any changes in network are propagated between routers. Routers placed on edge of area exchange ready paths between each other (with use of distance-vector approach).

The IS-IS is hybrid algorithm, that combines properties of distance - vector and link state algorithms. It was created for routing inside of autonomous systems.

3 Results

The analysis of the data placed in the Table 3.1 may serve as a summary of the research analysis performed on the implemented traffic control algorithms in the computer networks with the use of a custom simulator. In this table placed were the percentage values of fulfillment of a specific criterion for a given algorithm in comparison with the fulfillment of this criterion by the static routing. The data included in the table are based on the research performed on traffic generator feed in a simulator with the use of packet size histograms and their inter-arrival time. In table 3.1 in bold indicated are the best, whereas in italics second in value for a given criterion.

Tabela 3.1 Criteria deviation for algorithms in percentage

Criterion	RIP	IGRP	OSPF	IS-IS	EIGRP
loss ratio	-49,83%	-27,81%	*-57,14%*	**-59,52%**	-22,68%
allowed ratio	**54,18%**	14,42%	*51,09%*	50,00%	9,00%
utilization	**23,62%**	8,82%	*19,54%*	18,78%	10,22%
throughput	**54,76%**	15,72%	*51,29%*	50,22%	8,84%
queue length	7,33%	9,18%	**3,11%**	*4,07%*	11,63%
Power Formula	*35,14%*	4,75%	**36,52%**	29,76%	-1,15%
New Power Formula	206,92%	68,23%	*268,66%*	**289,64%**	53,58%
fairness	**0,03%**	*0,01%*	-0,20%	-0,18%	0,00%

When it comes to the null factor the best are IS-IS and OSPF algorithms. The reason for that may result from the fact, that every nod supported by those protocols has the knowledge of the whole network topology and on that basis it establishes the routes for the packets. It results in a greater loss of the packets connected with topological negotiations of the data bases, but in time it provides better route selection.

In the field of acceptability factor, meaning the amount of data reaching the destination, the best appears to be RIP, next in line is OSPF and IS-IS. This may be the result of the fact, that RIP does not use the network resources to such an extent as the link – state protocols do, which exchange far greater amounts of data between themselves.

Next criterion being a subject to research is the connection usage criterion. This criterion depends directly on the traffic occurring in the network being tested. It is well known, that in the real world no connection is used in 100%. Similar situation is reflected in the simulated example, which is based on both real topology as well as real traffic characteristics. For that particular reason examining the absolute values of the connection usage at the 10% level would discredit the algorithms, nevertheless it does not originate from the quality of the algorithm, but from the traffic generated by the generator. As it is shown in the Table 3.1, when it comes to the connection usage, the best is the RIP algorithm. The reason for that being the correlation of the criterion with the acceptability criterion and it is expressed in a different unit, as well as bandwidth aspect.

Bandwidth criterion is the third correlated measure (next to acceptability and connection usage). It provides the information about the number of packets reaching the destination in a period of time, thus it is connected with the previous criteria. It is not surprising for that reason, that the best in this category was RIP, followed by OSPF and IS-IS.

Queue length is the next criterion. All of the dynamic types of routing are worse in fulfilling of this criterion than the static routing. It is reflected by the positive values in the Table 3.1. This fact is caused simply by a significant (approximately 50%) increase of the packet number, which do not become null (null factor), thus they do not have to go through the network, which means they do not have to be placed in several queues. The smallest increase of the length of the queue was demonstrated by OSPF, followed by RIP.

The criterion based on the power equation allows to assess how well the network is perceived by the users, it is after all a functional criterion. This criterion, in its calculations, takes into consideration the delays in the data transfer, within the aspect of bandwidth and the arrival factor. For that reason the bigger the numerical value of this criterion, the better. In comparison to static routing, the best in fulfilling of this criterion was IS-IS, followed by OSPF. As can be seen, the connection – state type protocols after establishing the best routes cause shorter delays than the distance – vector ones, which are not aware of the whole topology, thus they have to find new routes while performing the operations, what causes delays.

New power equation is a modified criterion of the power equation, which unlike the first one depends on the number of the lost packets. This feature causes the correlation of the values it takes with the null factor values. That is why the best in the aspect of the new power equation is IS-IS, followed by OSPF, exactly as in the case of the null factor criterion.

The last criterion tested, is the integrity. The improvement of fulfillment is shown only by RIP and IGRP. As one can see, the protocols of the connection – state type, using their complicated metrics and appointing the best routes globally, push integrity aside. Those are not, however, significant differences, but merely below one percent.

It has to be pointed out, that the aforementioned relations and levels of fulfillment of given assessment criteria have been tested for a specific instance of topology and the traffic occurring within. Therefore, it cannot be unambiguously stated, that the most appropriate algorithm for this network, in respect of a given criterion, will fulfill this criterion better, regardless of the network it is used in and the type of network traffic it encounters.

As is known, the protocols of the connection – state type (IS-IS and OSPF) are characterized by a better scalability and are more commonly used in big networks. Besides, they return faster to the normal state after a failure, changes in topology or network initiation, than the distance – vector protocols, what was reflected in this research.

Protocols of the distance – vector type (RIP, IGRP) are of advantage in small, simple networks, where the inconveniences (as longer convergence) are redeemed by easy configuration and simplicity.

Hybrid protocols (here EIGRP), on the other hand, join the features of both connection – state and distance – vector protocols. They reach the convergence just as fast as the first type, but they are characterized by smaller demand for resources (just as distance – vector protocols). Despite the fact, that in theory this protocol should acquit itself best in all of the conditions, in the tested case its performance was the worst of all regarding almost every criteria. Nevertheless, as it was already mentioned, it is connected with the specification of the tested network.

4 Summary

In the series of two articles the authors presented the research commenced on the traffic control algorithms in the computer networks.

Within the first part presented were the popular algorithms and they were compared on the basis of the existing publications. The criteria, which may be used for the assessment of the algorithms, were described as well.

The second part provides the description of the works performed in the creation process of the author's simulator, allowing for commencing the research on the algorithms within the aspect of the presented criteria. The developed simulator may be extended with the implementation of some additional classes, responsible for simulating the traffic-control algorithms. There is also a possibility of implementation of supplementary statistics, which are to be gathered by the simulator, allowing to test the algorithms in the aspect of additional criteria. Since the simulator has been implemented with the use of the Java language, it is independent from the hardware and software, which allows for further development.

References

1. Cisco Systems Inc., (2004). A*kademia sieci Cisco. CCNA semestry 1 & 2*, Mikom.
2. Engineering Computer Network, Purdue University https://engineering.purdue.edu/ECN/
3. Hedrick C., (1988). *RFC 1058: Routing Information Protocol*, http://tools.ietf.org/html/rfc1058 Routgers University.
4. Kolbusz J., Paszczyński St., (2006). *Generator ruchu oparty na modelu „on-off”*, Materiały XXII KSTiT (Krajowego Sympozjum Telekomunikacji i Teleinformatyki).
5. Moy J., (1998). *RFC 2328: OSPF Version 2*, http://tools.ietf.org/html/rfc2328 Ascend Communications, Inc.
6. Oran D., (1990). *RFC 1142: OSI IS-IS Intra-domain Routing Protocol*, http://tools.ietf.org/html/rfc1142 Digital Equipment Corp.
7. Sommers J., Kim H., Barford P., (2004). *Harpoon: A Flow-Level Traffic Generator for Router and Network Tests*, Performance Evaluation Review 2004, Vol. 32, part 1, s. 382-393, ACM.
8. Tanenbaum A. S., (2003). *Computer Networks. Fourth Edition.*, Prentice Hall.
9. Wright R., (1999). *Elementarz routingu IP.,* Cisco Systems Inc., Mikom.
10. Barczak A., Sitkiewicz T., (2008). *Traffic Control algorithms in computer networks. Part 1. - Review of algorithms.* Studia Informatica