Journal of Theoretical and Applied Information Technology
15th November 2017. Vol.95. No 21

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

STUDYING NETWORK TRAFFIC USING NONLINEAR DYNAMICS METHODS

[1]SHARAFAT A. MIRZAKULOVA, [2]VYACHESLAV P. SHUVALOV, [3]ALEKSEY A. MEKLER

[1]Almaty University of Power Engineering and Telecommunications, Almaty, Kazakhstan
[2]Siberian State University of Telecommunications and Information Sciences, Novosibirsk, Russia
[3]Saint-Petersburg State Pediatric Medical University, Saint-Petersburg, Russia
E-mail: [1]mirzakulova@mail.ru; [2]shvp04@mail.ru; [3]mekler@yandex.ru

ABSTRACT

The development of the Internet, the constantly growing number of network users, and their mutual exchange of information is becoming an important communication bridge. However, this causes a series of technical difficulties, one of which is the growing requirements to network and server equipment and its maintenance. Therefore, the purpose of this study is to develop a computer program for teaching a neural network based on a computer network traffic table. A set of methods was used to achieve the set goal, including analysis, deterministic chaos, and systematization. The study used such software packages as TISEAN, MatLab, NetEmul, and Excel. The study generalized the experience that was relevant to the problem at hand. The study calculated the Lyapunov exponent, which characterizes the presence of chaos in the system. Analysis of the Lyapunov exponent enables using nonlinear dynamics methods to study the nature of the incoming and outbound traffic. With the help of the developed program, the neural network router is capable of prediction short-term parameters of a computer network; this information is sent to the system administrator, which will allow adapting the router to the estimated changes in the computer network.
Keywords: *Distribution series, Self-similarity, Chaotic processes, Phase portrait, Mutual information*

1. INTRODUCTION

The self-similarity problem was first mentioned by American scientists W. Leland, M. Taqqu, W. Willinger, and D. Wilson in 1993 [1]. The investigation of Ethernet traffic in the Bellcore network found that network traffic had the property of scale invariance, slowly decreasing dependencies (aftereffects), and could have high traffic burstiness (peak intensity of packet arrival for servicing to its mean value ratio) with a relatively low mean level. Studies on the structure of network traffic are ongoing [2].

Traffic is one of the most important indices in network monitoring [3]. The core of modern multiservice networks is based on a packet IP-network; however, the calculation methods have remained virtually the same as they were for the well-studied telephone networks with line switching. Traffic in multiservice networks is bursty, which increases the odds of overloading the network nodes, which causes buffer overflow. This is a problem for real-time applications (voice, video), since it causes lag, packet loss, and highly variable servicing time in network nodes.

This paper presents a study of network traffic using deterministic chaos methods, which is especially interesting for assessing the presence of chaotic dynamics in the process and its nature (share of deterministic chaos and random noise) during the processing and handling of data flows in a multiservice network. For instance, study [4] shows that nonlinear dynamics provides extensive methods for analyzing traffic; study [5] offers using nonlinear dynamics methods to determine the characteristic features of traffic; study [6] offers using said methods to identify anomalies in time series models; study [7] offers using said methods to predict time series. Study [8] shows that nonlinear dynamics methods and state models in a fractal phase space can be used to analyze the fractal properties of network processes.

When analyzing time series via methods used to study nonlinear dynamic systems, processes are considered chaotic and containing deterministic chaos [9]. From the perspective of linear methods of analysis, these processes are stochastic. However, they are not entirely random. In other words, a chaotic process is a cross between a deterministic process and a stochastic one.

Random and chaotic signals are similar in the time, spectral, and autocorrelation domains, which makes their identification difficult [9, 10]. At that, chaotic fluctuations, unlike stochastic ones, have an internal order, which manifests during the analysis of chaotic system motion in a phase space using nonlinear dynamics methods.

As was mentioned earlier, an important feature of a multiservice network is the uneven intensity of packet arrival at the communication network (bursty traffic), which is why studying the structure of network traffic is a relevant problem [6, 8]. These studies will enable solving problems related to improving the quality of servicing of real-time applications by network devices. These applications are critical to such parameters as losses, packet delay, and jitter. One of the numerical characteristics used in the dynamic chaos theory when studying time series – the Lyapunov exponent – determines the final predictability (the future is definitely determined by the past) [11]. Based on prediction, in case of an overload, respective processing algorithms of network devices (switches, routers) can prevent losses by redistributing the required capacity to the buffer of the port that is handling the frame, in the data field whereof the real-time application is. In addition, network devices that are used nowadays have port buffer capacities that are intended for handling the simplest flow, rather than a self-similar (fractal) one. The simplest flows are stationary, ordinary, and lack aftereffects (i.e., the flow can be regarded as a Markoff process).

The fractal theory serves as the basis for the quantitative description of dissipative structures that form in unbalanced conditions and that such structures form in multiservice networks during the transmission of information flows [12]. This is related to an uneven intensity of message arrival to a communication network that is based on packet switching, the growing number of users, the emergence and spread of the interaction technology in the "client-server" model network, the growing number of applications, and the transfer of real-time applications by means of packet technologies.

This study aims to develop a program for teaching neural networks.

2. METHOD

A set of methods relevant was used to achieve the aim, including analysis, synthesis, systems and logical methods, conceptualization, deterministic chaos, and the mathematical method. The study also generalized experience on the problem at hand.

Computation used procedures implemented in the TISEAN package, while routers were checked using the NetEmul package, as well as MatLab and Excel.

This study investigates the series of ranges of intervals between ARP (492 points in series) and MPEG (25,733 points in series) packets and the total measured traffic (278,556 points in series) of the backbone network traffic, measured using the Wireshark sniffer software (Figures 1, 2, and 3, respectively).

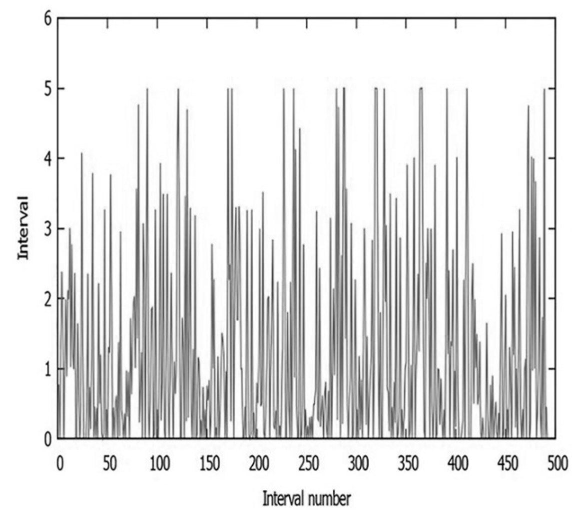


Figure 1: Distribution of intervals between ARP packets

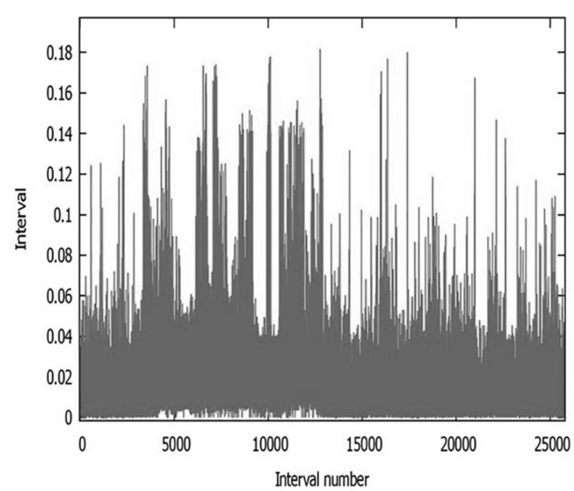


Figure 2: Distribution of intervals between MPEG packets

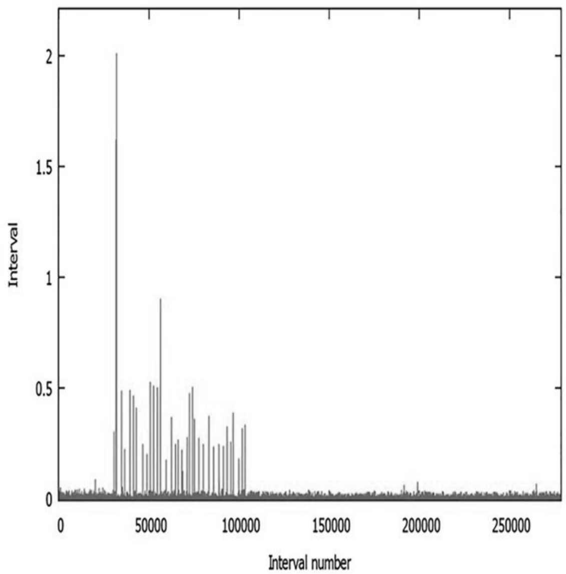


Figure 3: Distribution of intervals between packets (backbone traffic)

Study [13] used the IFS (Iterative Function System) clumpiness test to study noise: white (Figure 4), pink, etc.

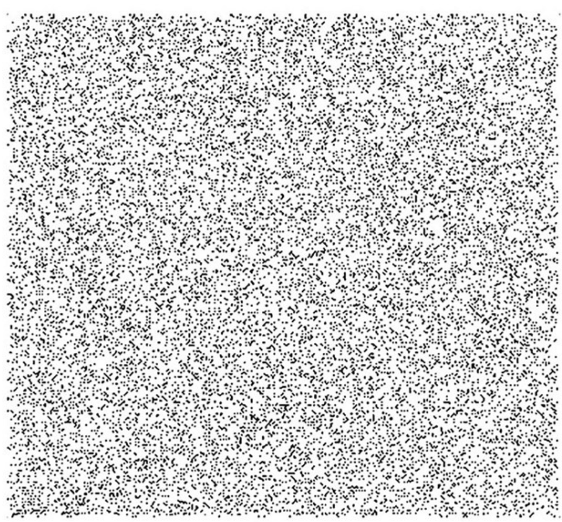


Figure 4: White noise

This resulted in a mapping of a one-dimensional set of points onto a plane. Depending on whether the filling of the space with points is homogenous or non-homogenous, the nature of the studied process can be assessed visually.

The ARP and MPEG series are studied using the IFS clumpiness test. First, the time series range of values is divided into quartiles. The first quartile corresponds to the lower left corner and further clockwise. The origin is placed in the center of the square; half the distance in the direction of the corner that corresponds to the quartile to which the first value of the studied signal appertains is plotted from the origin. Then, half the distance in the direction of the corner that corresponds to the quartile to which the next value of the studied signal appertains is plotted from the obtained point, etc. [13, 14]. The assessment of determinism is based on the fact that white noise fills the square evenly, deterministic processes cause diagonal structures to emerge, while chaotic processes fill the square unevenly [15].

Figures 5 and 6 show the filling of a set of points of studied ARP and MPEG series, respectively.

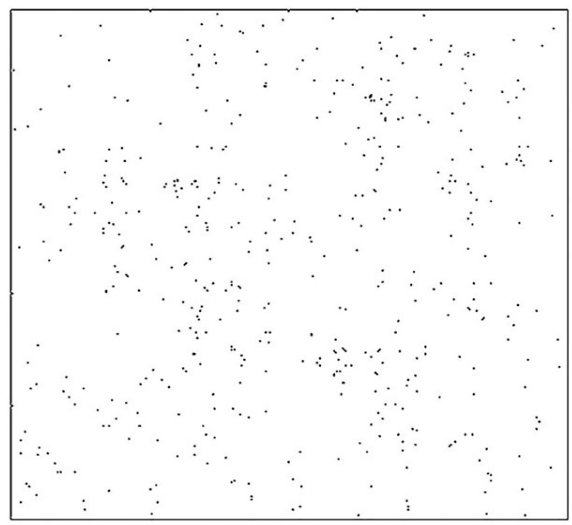


Figure 5: Result of processing with IFS clumpiness test (ARP)

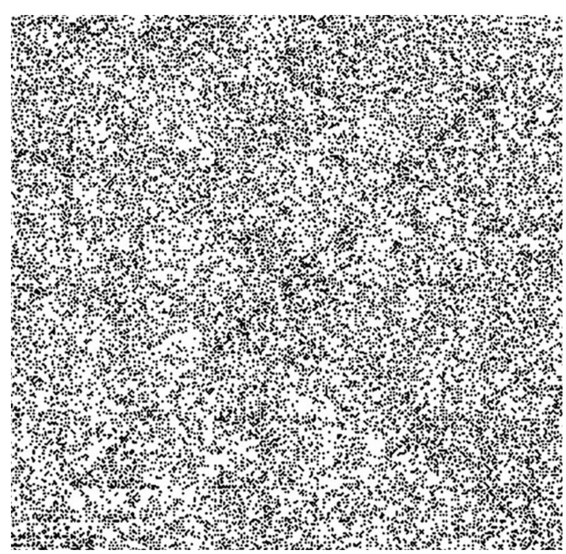


Figure 6: Result of processing with IFS clumpiness test (MPEG)

Unlike white noise, the studied ARP and MPEG series fill the square unevenly, which is indicative of either a quasiperiodic mode or chaotic dynamics in the studied processes, which is diagnosed based on the positive Lyapunov exponent. For instance, study [2] presents the following terminology:

- "noisy periodic mode" if the Lyapunov exponent is negative;
- "noisy quasiperiodic mode" if the Lyapunov exponent is close to zero;
- "noisy chaotic mode" if the Lyapunov exponent is positive.

Consider the dynamics of the studied series (ARP, MPEG) in the state space or phase space. The "delay" method was used to reconstruct the phase portraits of ARP and MPEG series (using the TISEAN software). In order to reconstruct the phase portrait of the time series, variable was plotted along the X-axis; the same variable, but with shift, was plotted along the Y-axis for the 2D format, and the same variable, but with a new axis and shift – for the 3D format [16, 17].

Figures 7, 8, 9, and 10 show the phase portraits of ARP and MPEG series in 2D and 3D formats, respectively.

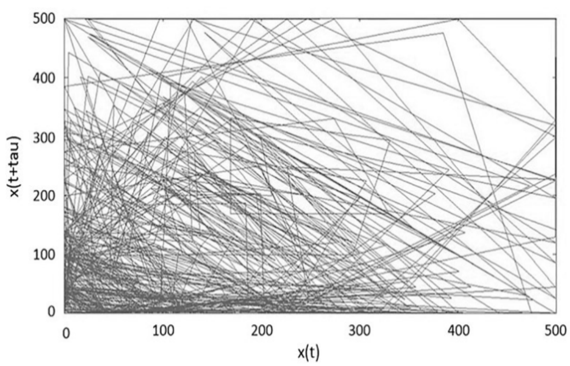


Figure 7: Phase portrait of ARP traffic in 2D

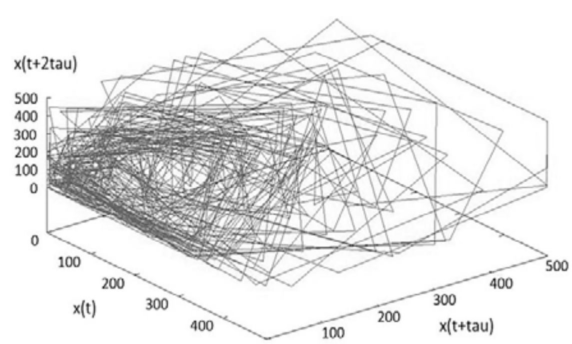


Figure 8: Phase portrait of ARP traffic in 3D

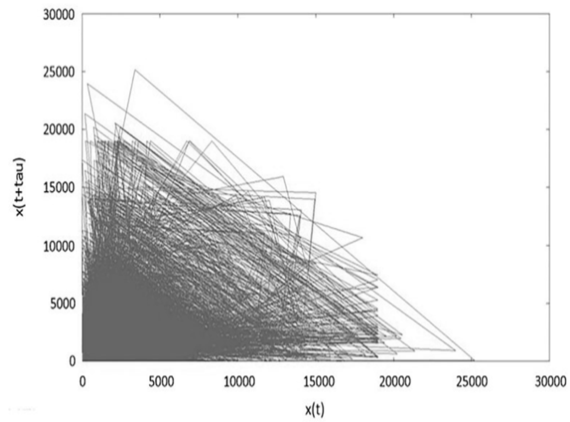


Figure 9: Phase portrait of MPEG traffic in 2D

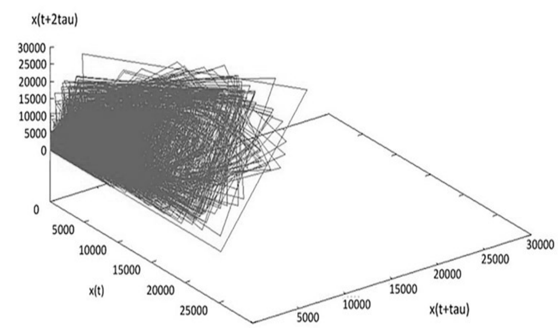


Figure 10: Phase portrait of MPEG traffic in 3D

Study [18] mentioned that the works of Grassberger, Procaccia [19], showed that a single time realization could be used to determine the correlation dimension and find out "how chaotic" the signal was, while the Kaplan-Yorke hypothesis linked the dimension-determined statistical structure of the attractor with the dynamics of motion in the attractor, connected by Lyapunov exponents [20].

The correlation dimension (D2) allows assessing the complexity of the system dynamics.

The plot in Figure 11, which shows the dependency of changes in the correlation dimension on the embedding dimension for the MPEG series [21], shows that the studied process has a saturation, which, however, is not pronounced. This means that the generative system is not random, but rather managed by a large number of parameters. The series is noisy at that.

Journal of Theoretical and Applied Information Technology
15th November 2017. Vol.95. No 21

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

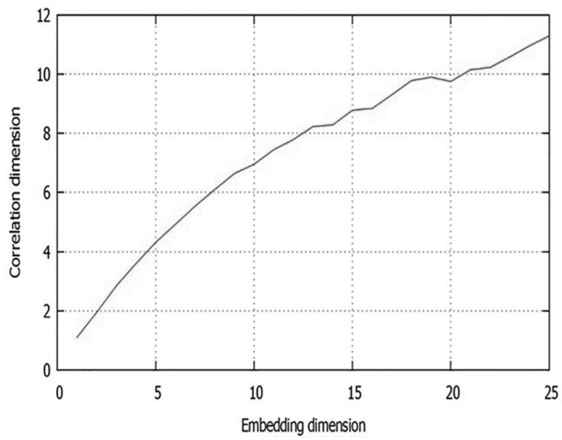


Figure 11: Dependency of changes in the correlation dimension on the embedding dimension (MPEG)

The embedding dimension of the MPEG series is 12, at which the correlation dimension is 7.79±0.35 and which is indicative of a certain saturation of the dependency, with regard to computational error.

Studies [22] showed that the strange attractor looks like a set of an infinite number of layers or parallel planes, the distance between which is infinitely small.

One of the main characteristics of a strange attractor is the sensitivity of its trajectories to the initial conditions. This means that two trajectories that are close to each other in the phase space at a certain initial point in time will diverge exponentially in a small mean time. At the same time, since the attractor has boundary dimensions, two trajectories in it cannot diverge infinitely. The exponential divergence-convergence of phase trajectories can be assessed using Lyapunov exponents. In order to identify nonlinear dynamic processes, it is necessary to calculate the largest Lyapunov exponent – . From the practical perspective, it is important that the Lyapunov exponents, while being invariants, can be calculated based on an experimentally obtained time series.

The largest Lyapunov exponent was calculated according to the algorithm developed in 1994 by H. Kantz [23]. Procedures implemented in the TISEAN software package were used for calculations [16, 17].

The advantage of this computation algorithm is that it does not require recapturing the attractor in the lag plane of the smallest required dimension, which exceeds the dimension of the initial attractor of the system. A small dimension of the lag space will suffice (in the TISEAN software package, the default lag space dimension is Demb=2). The index is calculated as the slope of the approximating line on a linear segment of the family of plots, obtained according to the Kantz algorithm, presented in [24nb]. The series is projected onto the trajectory in the lag space. Consider two neighboring points and co-directional trajectory fragments. One of them is denoted as, while the other point – the one the trajectory arrives at after a certain cycle. The distance between them, can be considered a small trajectory perturbation. Then, in l steps in the time series, one can obtain. If then the λ value can be considered the largest Lyapunov exponent. It is possible to calculate a family of functions:

$$S(\varepsilon, m, t) = \left\langle \ln\left(\frac{1}{|U_n|} \sum_{s_{n'} \in U_n} |s_{n+t} - s_{n'+t}| \right) \right\rangle_n \quad (1)$$

where m is the lag space dimension, set as a parameter in calculations

(usually set in calculations by m=2);

is the set of trajectory points;

t is time ( );

is the sampling frequency.

Other parameters set for the calculations are the lag value and the Theiler window value.

The lag value is set equal to the coordinate of the first local minimum of the mutual information functions; the Theiler window is set equal to the first general minimum of the space-time separation plot [21].

Thus, the sequence of actions for calculating the Lyapunov exponent is as follows:

- calculation of the lag using the mutual information function that reflects both the linear and nonlinear relationship between the two variables: in this case, two values of the same variable with a certain lag between them are used instead of two variables and the mutual information functions is built from the lag value;
- calculation of the Theiler window using space-time separation plots (the previously obtained lag value is used as a parameter in this procedure);
- calculation of Lyapunov exponents using the previous obtained lag and Theiler window values as parameters.

Procedures implemented in the TISEAN software package were used for calculations [16, 17].

Figures 12 and 13 shows the dependencies of mutual information I on lag for the ARP series and the general series of observation. In further calculations, the lag value is set equal to the coordinate of the first minimum of the obtained mutual information function. Since the solution of this problem requires only the location of the first local minimum of the functions, rather than its absolute values, the TISEAN package used an

Journal of Theoretical and Applied Information Technology
15th November 2017. Vol.95. No 21

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

algorithm that does not provide its real value, but accelerates calculations significantly.

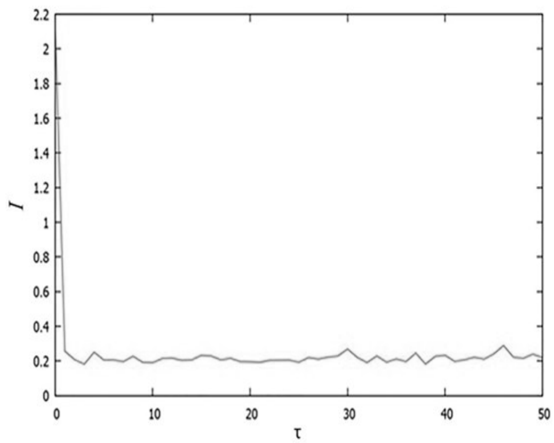


Figure 12: Dependency of mutual information I on lag (ARP)

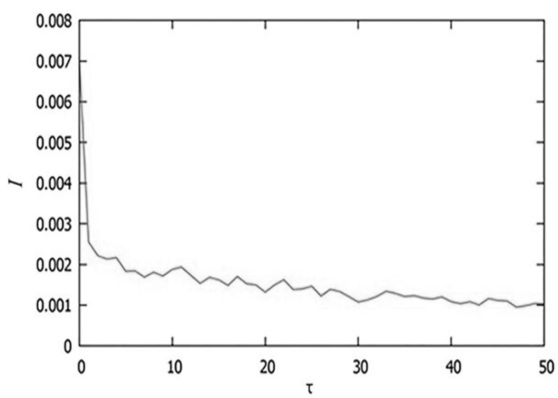


Figure 13: Dependency of mutual information I on lag (general series)

In order to avoid errors caused by an insufficient extent of observation series, J. Theiler set forth a modification of the computation algorithm, in which he offered to exclude the pairs of points, the distance between which in the initial time series was less than w steps, since the presence of correlated points causes computational errors . The w value is called in Theiler window. The optimal size of the Theiler window is achieved using a space-time separation plot that is a simple equal-density curve of the probability that two points in the time series, located thereon at distance will find themselves in the restored attractor at a distance that does not exceed. At that, a family of curves that correspond to various values of probability density is built.

Figures 14 and 15 shows the families of space-time separation plots for probability densities of ARP series and the general observation series, respectively.

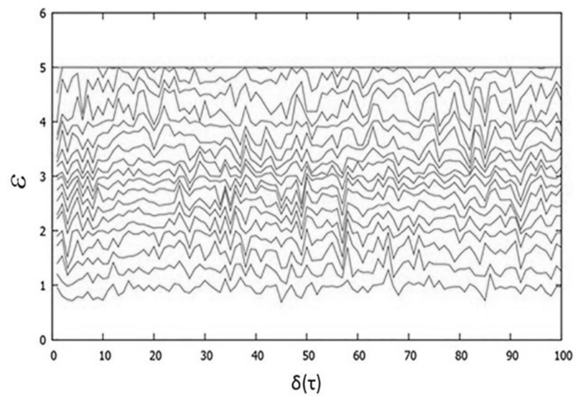


Figure 14: Dependency plots of relative distance on ε-neighborhood (ARP)

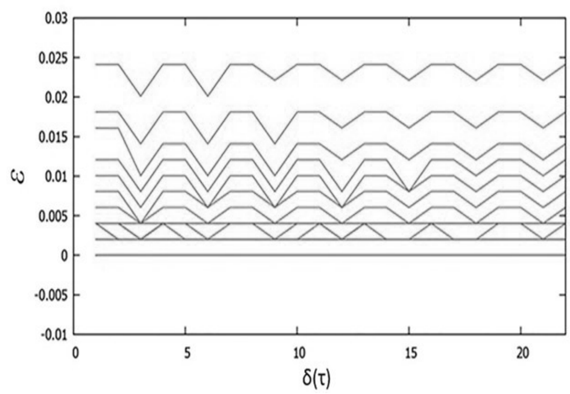
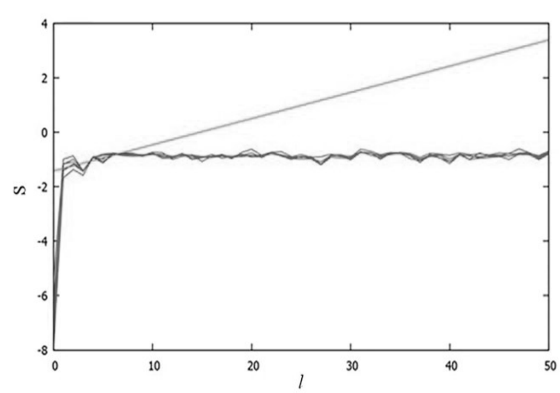


Figure 15: Dependency plots of relative distance on ε-neighborhood (general series)

In the studied time series (ARP and general), the above dependency plots of relative distance on ε-neighborhood have common first minimums that equal 3.

Figures 16 and 17 show the dependency curves for the ARP series and the general series of observation, respectively.



Journal of Theoretical and Applied Information Technology
15th November 2017. Vol.95. No 21

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 16: Dependencies for the ARP series

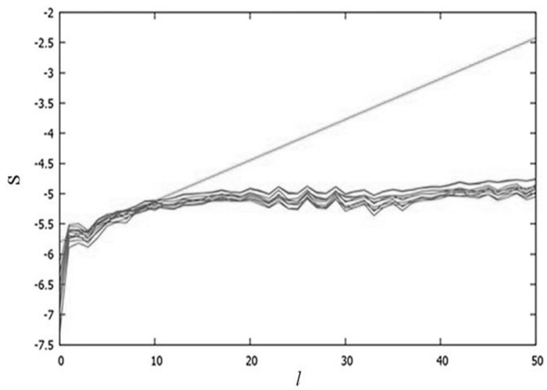


Figure 17: Dependencies for the general series of observation

The analysis of the above plots shows that the linear segment is set in the interval [1, 4] for the ARP series. The line slope is 0.0962692 ± 0.02242, which corresponds to the Lyapunov exponent value.

For the general series of observation, the linear plot is set in the interval [2, 9]. The line slope is 0.0361674 ± 0.003652, which corresponds to the Lyapunov exponent value.

Within the attractor, insignificant changes in initial conditions may cause significant changes in the evolution of the system; therefore, the Lyapunov exponent can be a measure of how significant such changes are going to be. The more sensitive the system is to initial conditions, the larger this exponent will be.

3. USING THE NEURAL NETWORK APPROACH TO STUDY NETWORK TRAFFIC

Mathematical models of neural networks are a convenient computational mechanism with the following important characteristics:

- learning capability;
- adaptability;
- high parallelism and computation speed;
- predictable computational robustness, i.e. the ability to find accurate solutions with inaccurate input data.

Neural network algorithms are determined by the neural setting of the routing problem (its neural interpretation, to be more precise) and an appropriate model of a neural network that solves the problem in this setting.

The creation of neural computers and the modeling of adaptive neural networks are considered the most promising area for the solution of many problems related to artificial intelligence. A neural network used in the routing algorithm is a mathematical model of parallel computations, which contains simple interacting processing elements – artificial neurons. The advantage of neural networks over conventional algorithms is their learning capability. This study uses the learning principle that includes a teacher and the Widrow-Goff algorithm.

Developing a program for teaching neural networks based on a computer network traffic table created in Microsoft Excel

Assume the structure of the local computer network is presented in the form of a graph (Figure 2). Marks are set along the edges of the graph according to the following attributes: minimum route edges are assigned the (1) value, while other edges are assigned the (-1) value. For instance, the route from node 1 to node 2 equals 15 and is coded with one.

$$\beta 12 = 15 = \succ 1 \qquad (2)$$

or instance, possible routes between nodes 1 and 3 can be coded as follows:

$$\begin{aligned} &\mu_{13} = \beta_{12} \to \beta_{23} = 1 \to 1, \\ &\mu_{13} = \beta_{14} \to \beta_{42} \to \beta_{23} = -1 \to -1 \to -1, \\ &\mu_{13} = \beta_{14} \to \beta_{45} \to \beta_{53} = -1 \to 1 \to -1, \end{aligned} \qquad (3)$$

where μ_{ij} is the total distance between network nodes presented in the form of codes, β_{ij} is the coded distance between network nodes (according to the conditions of the network administrator, this parameter can also express the data transfer speed, lag, etc.).

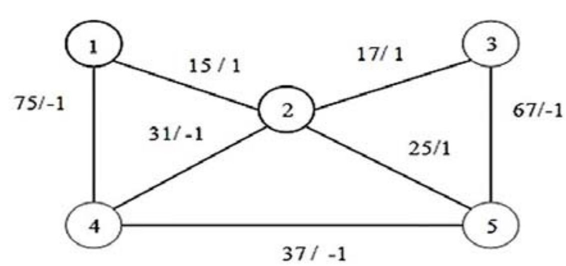


Figure 18: Graph with marks

The neural network chooses the route that is coded with a sequence of positive ones, where (1) is the right route, while (-1) is the wrong route. In order to represent the route graph, a neural network structure is chosen, where X_i are inputs of the neural network, Y_j are outputs of the neural

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

network, where i=1..N; j=1..M; N=M=5, x_i is the current value of the input signal, and y_j is the current value of the output signal.

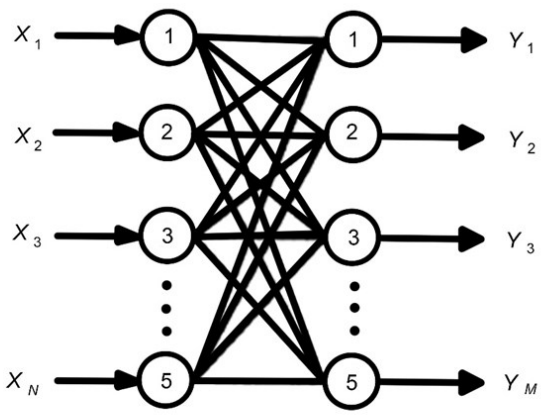


Figure 19: Neural network structure

Consider a coded table of optimal routes for the set neural network (Table 1).

Table 1: Optimal graph routes (Figure 2) for the neural network

	y_1	y_2	y_3	y_4	y_5
x_1	-1	1	-1	-1	-1
x_2	1	-1	1	-1	1
x_3	-1	1	-1	-1	-1
x_4	-1	1	-1	-1	1
x_5	-1	1	-1	1	-1

Consider a neural network with a signature activation function *SGN* [9].

$$Y_i = SGN\left(\sum_{i=1}^{N} W_{ij} X_i\right) = \begin{cases} 1 \text{ при } \sum_{i=1}^{N} W_{ij}\, X_i \geq 0 \\ -1 \text{ при } \sum_{i=1}^{N} W_{ij}\, X_i < 0 \end{cases} \quad (4)$$

where W_{ij} is the weight factor of the neural network.

Assume the neural network input is receiving coded signals, which are routes:

$$x_1^T = (-1\;1\;-1\;-1\;-1);$$
$$x_2^T = (1\;-1\;1\;-1\;1);$$
$$x_3^T = (-1\;1\;-1\;-1\;-1);$$
$$x_4^T = (-1\;1\;-1\;-1\;1);$$
$$x_5^T = (-1\;1\;-1\;1\;-1), \quad (5)$$

The weight factors of the neural network matrix for images $x_i = (i = 1, 2, 3, 4, 5)$ are found according to the following formula:

$$W_i = x_i\, x_i^T, \quad (6)$$

$$W_1 = x_1\, x_1^T = \begin{pmatrix} 1 & -1 & 1 & 1 & 1 \\ -1 & 1 & -1 & -1 & -1 \\ 1 & -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 & 1 \end{pmatrix},$$

$$W_2 = x_2\, x_2^T = \begin{pmatrix} 1 & -1 & 1 & -1 & 1 \\ -1 & 1 & -1 & 1 & -1 \\ 1 & -1 & 1 & -1 & 1 \\ -1 & 1 & -1 & 1 & -1 \\ 1 & -1 & 1 & -1 & 1 \end{pmatrix},$$

$$W_3 = x_3\, x_3^T = \begin{pmatrix} 1 & -1 & 1 & 1 & 1 \\ -1 & 1 & -1 & -1 & -1 \\ 1 & -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 & 1 \end{pmatrix},$$

$$W_4 = x_4\, x_4^T = \begin{pmatrix} 1 & -1 & 1 & 1 & -1 \\ -1 & 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & 1 & -1 \\ -1 & 1 & -1 & -1 & 1 \end{pmatrix},$$

$$W_5 = x_5\, x_5^T = \begin{pmatrix} 1 & -1 & 1 & -1 & 1 \\ -1 & 1 & -1 & 1 & -1 \\ 1 & -1 & 1 & -1 & 1 \\ -1 & 1 & -1 & 1 & -1 \\ 1 & -1 & 1 & -1 & 1 \end{pmatrix}. \quad (7)$$

The resulting network weight matrix W is calculated using the following formula:

$$W = \sum_{i=1}^{5} W_i = \begin{pmatrix} 5 & -5 & 5 & 1 & 3 \\ -5 & 5 & -5 & -1 & -3 \\ 5 & -5 & 5 & 1 & 3 \\ 1 & -1 & 1 & 5 & -1 \\ 3 & -3 & 3 & -1 & 5 \end{pmatrix}. \quad (8)$$

In order to determine the reference image, saved in the memory, it is necessary to calculate the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

product of $WX_i = Y_i\ (i = 1, 2, 3, 4, 5)$. For instance, with an x_3 entry, we have

$$Wx_3 = \begin{pmatrix} 5 & -5 & 5 & 1 & 3 \\ -5 & 5 & -5 & -1 & -3 \\ 5 & -5 & 5 & 1 & 3 \\ 1 & -1 & 1 & 5 & -1 \\ 3 & -3 & 3 & -1 & 5 \end{pmatrix} \times \begin{pmatrix} -1 \\ 1 \\ -1 \\ -1 \\ -1 \end{pmatrix} = \begin{pmatrix} -19 \\ 19 \\ -19 \\ -7 \\ -13 \end{pmatrix}. \tag{9}$$

Considering the activation function (2), we obtain the output signal $(-1\ 1\ -1\ -1\ -1)$, i.e. the neural network restored the saved x_3 image correctly. Similar results were obtained for other images.

Assume an x_3 image damaged in the second bit is entered into the network input and this image imitates the reduction of quality of the test signal transmitted from the nearest computer network node.

$$\bar{x}_3^T = (-1\ -1 -1\ -1\ -1). \tag{10}$$

The application of the W matrix to $\bar{x}_3^T$ produces the following:

$$Wx_3 = \begin{pmatrix} 5 & -5 & 5 & 1 & 3 \\ -5 & 5 & -5 & -1 & -3 \\ 5 & -5 & 5 & 1 & 3 \\ 1 & -1 & 1 & 5 & -1 \\ 3 & -3 & 3 & -1 & 5 \end{pmatrix} \times \begin{pmatrix} -1 \\ 1 \\ -1 \\ -1 \\ -1 \end{pmatrix} = \begin{pmatrix} -11 \\ 1 \\ -11 \\ -7 \\ -9 \end{pmatrix}. \tag{11}$$

The network restored the damaged image correctly

Microsoft Excel software is used to simplify the calculation (Figure 20 for an ordinary signal and Figure 21 for a distorted signal).

ΣW =	-5	5	-5	-1	-3
	5	-5	5	1	3
	1	-1	1	5	-1
	3	-3	3	-1	5
	-19	=	-1		
	19	=	1		
Wx3 =	-19	=	-1		
	-7	=	-1		
	-13	=	-1		

Figure 20: Weight factor calculation program (set input values)

	5	-3	5	1	3
ΣW =	-3	5	-3	1	-1
	5	-3	5	1	3
	1	1	1	5	-1
	3	-1	3	-1	5
	-11	=	-1		
	1	=	1		
Wx3 =	-11	=	-1		
	-7	=	-1		
	-9	=	-1		

Figure 21: Weight factor calculation program (distorted input values)

Thus, the offered approach can be successfully used in computer network router software.

4. DISCUSSION

A computer system expert plays the key role in the creation of neural networks. The expert makes corrections during the teaching and testing [7] of input and output parameters of the routing algorithm.

Another advantage of neural networks, as mentioned before, is their ability to extrapolate data and predict them in a short term. Neural networks based on accumulated data determine the analytical dependency (linear or nonlinear) and use it for interpolation and extrapolation. Neural networks also have the ability of prediction, which enables predicting the estimated data delay between computer network nodes at a set point in time and choosing the optimal information transmission route with time lead based on prediction data.

The following is a second emulation experiment to test this assumption.

Assume the router remembers the delay between nodes 1 and 2 and saves these data in its memory. For instance, at a certain time, the delay is

Journal of Theoretical and Applied Information Technology
15th November 2017. Vol.95. No 21

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4 seconds during the first day, 6 seconds during the second day, 7 seconds during the third day, etc. (Table 2).

Table 2: Dependency of traffic on the day of observation

Day (*P*)	1	2	3	4	5	7	8	9	10
Computer network traffic (GB)	4	6	7	9	10	13	14	16	17

After implementing the network in MatLab (Neuro Toolbox) [10] and teaching the neural network, it is expedient to assess the adequacy of its operation by comparing real and estimated values. The set of commands for creating the neural network is as follows:

>> *p* = [1 2 3 4 5 6 7 8 9 10];
>> *t* = [4 6 7 9 10 11 13 14 16 17];
>> net = newlind(p,t);
>> *y* = sim(net,p)

Network check:

y = 4.2909 5.7152 7.1394 8.5636 9.9879 11.4121 12.8364 14.2606 15.6848 17.1091.

The result of teaching demonstrated an adequate match. The MatLab (Simulink) application [10] is used to enable prediction capability and additional modification of the neural network (Figure 6).

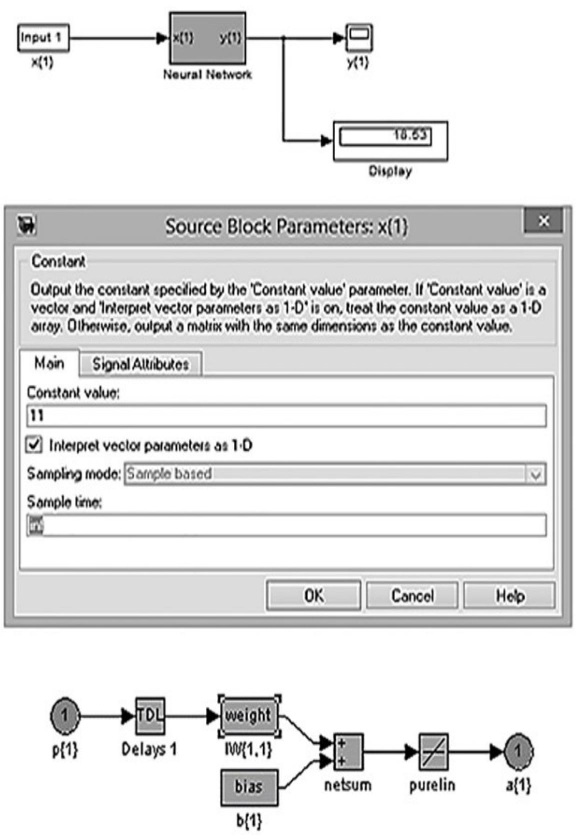


Figure 22: Simulation model of the neural network in MatLab (Neuro Toolbox)

The results of the extrapolation of the delay parameter onto day 11 is 18.5 seconds. Thus, the neural router is capable of successfully predicting short-term computer network parameters based on accumulated data. The predicted information is calculated on the system administrator server.

The assumption is that during operation, the routers will be sending information (tables of traffic and delay time between nodes) to the system administrator server. This also allows adapting the router to the estimated changes in computer network parameters, thus improving its performance. Thus, neural routers that are based on the algorithm of back propagation of error are capable of quickly determining the total delay between nodes in a wide range of changes in computer network parameters. Considering this capability, the router can successfully adapt to various changes in computer network parameters. A manufacturing local computer network (Figure 22) was designed to test the offered neural network approach and compare it with the regular adaptive routing algorithm (distance vector). The emulation and verification of the temporal capabilities of routers was carried out using the NetEmul package [11].

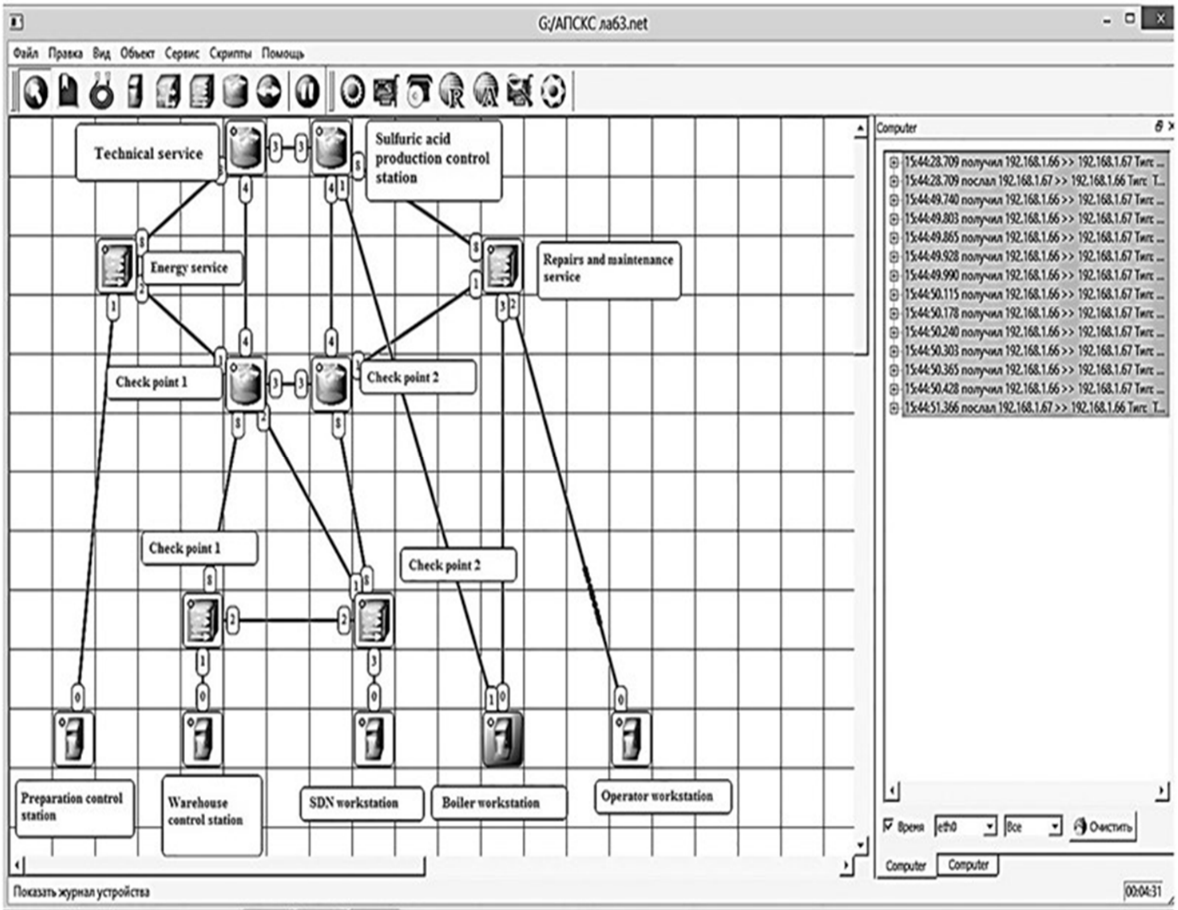


Figure 23: Simulation modeling for the comparative analysis of the effectiveness of routing methods using the NEMETUL package, by the example of a manufacturing local computer network

The comparative characteristics of the operation of two algorithms are presented in Table 3.

Table 3: Comparative analysis of the characteristics of a regular and neural network traffic prediction algorithm

Criteria	RIP algorithm	Neural algorithm
Ability to adapt	+	+
Optimal route estimation time, s	5	2
Correction of distorted signal	-	+
Simple implementation	+	+
Ability to predict network parameters	-	+
150 Kbit packet delivery time (s)	15	7

5. CONCLUSION

The study investigated network traffic using deterministic chaos methods. The IFS clumpiness test shows that the studied series fill the space unevenly, i.e. they differ from white noise and have localized clusters.

The study of the correlation dimension of the MPEG time series and general series of observation showed that the studied processes were noisy and their generative system was not random, but rather managed by a large number of parameters.

The MatLab environment was used to test a neural network router. Thanks to the accumulated data, the router was able to predict short-term parameters of the computer network. This

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

information will be calculated on the system administrator's server. This function will enable adapting the router to the estimated changes in the computer network, thus improving its performance.

The calculated Lyapunov exponents for the time series are positive, which is indicative of the presence of chaos in the dynamic system. Thus, nonlinear dynamics methods can be used to study the nature of incoming and outbound traffic by analyzing the Lyapunov exponent.

REFRENCES:

[1] Leland W, Taqqu M, Willinger W, Wilson D. On the self–similar nature of Ethernet traffic. IEEE ACM T. Network. 1994;2(1):1–15.

[2] Kuznetsov AP, Kuznetsov SP, Sedova YuV. On scaling properties under the effect of noise in circle mapping with a rotation number, set as a gold mean. Higher Educational Institution News. 2005;13(5-6):56-57.

[3] Fedorova ML, Ledeneva TM. The self-similarity of multiservice network traffic. Voronezh State University Bulletin, Systems Analysis and Information Technologies. 2010;1:46-54.

[4] Petrov VV. Statistical analysis of network traffic. MPEI National Research University. Moscow: Institute of Radio Engineering and Electronics; 2003.

[5] Edemskaya YeN, Belkov DV. Studying video traffic using nonlinear dynamics methods. Donetsk National Technical University Studies, 2013;2(18):10-17.

[6] Ternovoy VP. Developing methods of control and diagnostics of distributed systems of railway automation based on the analysis of information flows, 20. Rostov-on-Don: Dissertation abstract; 2013.

[7] Petrov VV. The structure of teletraffic and the algorithm for ensuring the quality of servicing under the self-similarity effect, 199. Moscow: Moskow Energy Resources Institute; 2004.

[8] Zaborovsky VS, Mulyukha VA, Podgursky YuYe. Computer networks and telecommunication. Modeling and analyzing computer networks: the telematics approach. SPb: Saint Petersburg State Polytechnic University; 2010.

[9] Skril DYu. Intelligent identification of chaotic processes and signals; 2013.

[10] Filatov A. Chaos in technical systems. Criteria for detecting chaos; 2031.

[11] Malinetsky GG, Kurdyumov SP. Nonlinear dynamics and prediction problems. Proceedings of the Russian Academy of Sciences. 2001;71(3):210-232.

[12] Belkov DV. Studying network traffic. Computer Science, Cybernetics and Computer Engineering. Donetsk: Donetsk National Technical University; 2009.

[13] Sprott JC, Rowlands G. Chaos data analyzer; the professional version. New York: AIP; 1995.

[14] Aks DJ. 1/f Dynamic in Complex Visual Search: Evidence for Self-Organized Criticality in Human Perception. Tutorials in contemporary nonlinear methods for the behavioral sciences. 2005;326-359.

[15] Sychev VN. Using the framework of nonlinear analysis of dynamic systems to process experimental data. Bulletin of Kyrgyz Russian Slavic University. 2011;11(4).

[16] TISEAN. V2.1 software package for analyzing the time realization using nonlinear dynamics methods: 2000. [cited 2016 May 25]. Available from: http://www.mpipks-dresden.mpg.de/~tisean/TISEAN_2.1/index.html

[17] Hegger R, Kantz H, Schreiber T. Practical implementation of nonlinear time series methods: The TISEAN package. Chaos. 1999;9:413.

[18] Schuster G. Deterministic chaos. Moscow: Mir;1988.

[19] Grassberger P, Procaccia I. Characterization ofstrange attractors. Phys. Rev. Lett. 1983;50:346-349.

[20] Frederickson P, Kaplan J, Yorke J. The Lyapunov dimension of strange attractors. J. Different. Equat. 1983;49(2):185–207.

[21] Mekler AA. Calculation of EEG correlation dimension: Large massifs of experimental data. Computer Methods and Programs in Biomedicine. 2008;92:154-160.

[22] Veres A, Boda M. The Chaotic Nature of TCP Congestion Control. [cited 2016 May 25]. Available from: http://mechatronics.ece.usu.edu.

[23] Kantz HA. Robust method to estimate the maximal Lyapunov exponent of a time series. Phys. Lett. A. 1994;185:77.

[24] Kennel MB, Brown R, Abarbanel HDI. Determining embedding dimension for phase-space reconstruction using a geometrical construction. Phys. Rev. A. 1992;45:3403.