Applied problems of information systems operation

UDC 004.75 **doi: 10.20998/2522-9052.2020.4.18**

Viacheslav Davydov, Daryna Hrebeniuk

National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine

DEVELOPMENT THE RESOURCES LOAD VARIATION FORECASTING METHOD WITHIN CLOUD COMPUTING SYSTEMS

Abstract. The **subject** of research in the article is the models and methods of resources load forecasting in cloud computing systems using the mathematical apparatus of neural networks. The **aim** of the work is increasing the efficiency of computing systems resources usage (such as RAM, disk space, CPU, network) by developing methods of resources load forecasting. The article addresses the following **tasks**: development of an integrated approach to the problems of resources load forecasting within cloud computing systems, which includes the synthesis of a combining forecasting neural network; development of a forecasting neural network model based on Elman neural network; development of a method for training a neural network based on an artificial immunity algorithm; evaluation of the effectiveness of the developed method. To solve the set tasks, the approaches and **methods** of artificial neural and immune systems were used, as well as methods of theoretical research, which are based on the scientific provisions of the theory of artificial intelligence, statistic, functional and systemic analyzes. The following **results** were obtained: on the basis of the analysis of resources load forecasting methods in cloud computing systems, the main results of the methods were revealed, the advantages and disadvantages were demonstrated. On the basis of the research results analysis, the necessity of improving analytical methods for forecasting the load has been proved. The method of computing resources load forecasting in cloud computing systems has been improved, which makes it possible to obtain more accurate assessment results and prevent overloads in cloud computing systems. The results obtained are confirmed by the experiments carried out using the means of the infrastructure of private infrastructure services. **Conclusions**: improved the resources load forecasting method based on the mathematical apparatus of artificial neural networks to improve the efficiency of their usage.

Keywords: Elman neural network; cloud computing systems; resources load forecasting.

Introduction

In the modern world, the volume of data being operated is increasing. Initially enterprise programs were deployed on a physical server and vertical scaling was used for them. As a result of this physical server scaling, physical resources were added (for example, RAM, HDD, CPU). However, this approach to scaling has come across the problem of the motherboard and processor frequency physical limitation. The next stage in the development of enterprise application administration was horizontal scaling. At the same time, the clustering approach was used for servers. This approach was a short-lived success, as the amount of resources operated was used inefficiently. In addition, with a lack of resources, it was necessary to quickly buy a new hardware, which was not always possible.

An alternative approach to the such systems administration development was the presentation of the cloud computing systems development. Giant corporations such as Amazon, Microsoft have made their data centers in which you can rent the necessary resources amount. The volume of resources to be operated varies almost instantaneously. And at present, this approach is a trend. However, it is increasingly problem for vendor companies to efficiently allocate resources of cloud computing systems (CCS) between virtual machines. One approach to solving this problem is to forecast load change. In turn, this should avoid unwarranted migration of virtual machines during a short-term server overload.

Analysis of recent research and publications. Existing forecasting methods are divided into five main groups [4]:

1) heuristic forecasting methods;
2) mathematical methods of temporal and spatial extrapolation;
3) methods of modeling development processes;
4) logical and structural methods of artificial intelligence.

For a similar task, article [1] described the solution to the problem of predicting resource consumption in CCS using the exponentially weighted moving average (EWMA) method. However, this method, like other statistical forecasting methods, allows for significant simplification and does not allow identifying implicit patterns, which leads to low accuracy of forecasts when analyzing the complex systems behavior.

Based on the principle of "necessary diversity" by W. R. Ashby [2], since the cloud system is a complex system [5] with an extensive set of indicators, methods that take into account the full variety of available parameters are needed to predict its behavior [6]. In this regard, the application of a forecasting model based on artificial intelligence methods, in particular, artificial neural networks, is justified. This networks have proven to be good for predictive tasks by the following reasons:

1) the ability of artificial neural networks to carry out multivariable forecasting taking into account the urgency of forecasted processes;
2) artificial neural networks forecasting speed, achieved by maximum parallelism of the information processing;
3) insensitivity to lack of a priori dynamics information compensated by precedent information;
4) ability to process data presented in different scale types by reducing to a logical scale without compromising predictive speed;
5) ability to solve poorly formalized problems by

identifying implicit analogies of observational protocol precedents;

6) "Holography" or the ability to preserve properties when a randomly selected part of artificial neural networks is destroyed due to complete connectivity and a large number of artificial neurons;

7) ability for further training. Input information can be used to further train the forecast model without having to modify it for new conditions adapting [10];

8) possibility of forecasting jumps and events not previously observed in the training sample of the observed object.

Among the variations of neural networks, Elman networks should be distinguished as promising [1] when using resource allocation and their forecasting in CCS.

For training the Elman network, the reverse error propagation method is traditionally used. However, this method is inevitably characterized by a high level of error, which significantly reduces the accuracy of the neural network forecast [12].

Another common approach is to use genetic algorithms to match optimal network weights. The use of genetic algorithms for Elman network training is described in detail in [1]. However, this approach has two drawbacks: firstly, for a genetic algorithm, the probability of finding a locally significant solution is high, this can lead to the incorrect forecast; secondly, the genetic algorithm is quite resource intensive and cloud computing system indicators entire set analysis can lead to a time increase in the significant calculation of weights for each network [11], which again can lead to an unacceptable delay in forecasting the functioning parameters of the CCS and by this way decrease the resource allocation process efficiency [12].

A variety of evolutionary algorithms are artificial immune systems (AIS). Among the AIS advantages, it should be noted high execution speed, which is especially relevant for the solved problem due to the large number of simultaneously analyzed parameters [10], as well as a lower probability of finding locally optimal values compared to genetic algorithms. Artificial immune systems are successfully used for machine learning [6-8].

The aim of the article. In order to effectively balance the CCS load, it is necessary to evaluate how it will change over time, since, based only on the current load, it is quite difficult to prevent a lack of resources - we can only eliminate it in fact. Moreover, in some situations, the virtual machines migration is not required and only increases the cost of computing resources for migration. Literature analysis showed the promise of using neural networks to the CCS load change forecast. Thus, the purpose of the work is to improve the existing forecasting algorithms load variation based on the Elman neural network.

Materials and methods

Solving the problem of the predicting load variation method in CCS using artificial neural networks is divided to the following stages [7]:

– initial data collection and presentation in a single form in precedents table;

– synthesis of forecasting neural network architecture;

– forecasting model synthesis by the neural network train on training sample situations;

– obtaining a forecast for the required advance period;

– forecast model verification according to the selected criterion.

We need to use a set of statistics to predict the actual state of CCS resources over a time interval [10].

The forecasting neural network model should be able not only to continuously process the CCS technical state (TS) dynamic parameters large number [10], forecast background factors, but also to take into account heterogeneous information about the current and planned operating modes of the cloud computing system. The neural network forecasting system, in turn, should take into account information about the system logic, as well as expert information. To solve the problem of forecasting, it is necessary to take into account a large range of parameters that can be decomposed into the following types [8]:

– information about the object TS previous dynamics;

– information about the object forecast background dynamics;

– information about object elements reliability;

– expert information;

– morphological information (information about precedents);

– additional information about operation logic of the object and its elements.

At the same time, the input values are a set of indicators and characteristics for each CCS host over the last day with a time interval of 5 minutes. This value was selected because the specified time interval is usually sufficient to perform a virtual machine live migration. Such characteristics include:

- Static host:
  - ▪ CPU characteristics I_{CPU}: F_{CPU} - CPU clock frequency, N_{CPU} - CPU number, N_{core} CPU cores number, V_{cache} - L2 cache capacity, F_{MIPS} - count of millions integer operations per second, F_{MFLOPS} count of millions floating point operations per second.
  - ▪ RAM characteristics I_{mem}: V_{mem} - RAM capacity, F_{mem} - RAM clock frequency, $N_{channel}$ - RAM channel; Disk subsystem features I_{disk} (drive type); Network subsystem characteristics I_{net} (network bandwidth).
- Host dynamic characteristics (load indicators):
  - ▪ CPU load for *i* core $L_{CPU}^{i}:\{L_{CPU_{t=1}}^{i},L_{CPU_{t=2}}^{i},..,L_{CPU_{t=n}}^{i}\}$;
  - ▪ RAM load $L_{mem}:\{L_{mem_{t=1}},..,L_{mem_{t=n}}\}$;
  - ▪ disk subsystem load $L_{disk}:\{L_{disk_{t=1}},..,L_{disk_{t=n}}\}$;
  - ▪ disk subsystem response time $T_{disk}:\{T_{disk_{t=1}},T_{disk_{t=2}},..,T_{disk_{t=n}}\}$;

- network load on each *j*-th network interface $L_{net}^{j}:\{L_{net_{t=1}}^{j},L_{net_{t=2}}^{j},..,L_{net_{t=n}}^{j}\}$;
- network response time on each *k*-th network interface $T_{net}^{k}:\{T_{net_{t=1}}^{k},T_{net_{t=2}}^{k},..,T_{net_{t=n}}^{k}\}$.

The output values are dynamic load indicators forecast values.

Cloud computing systems load modeling using a hybrid approach. Obtaining an accurate forecast in a computational environment with dynamically changing parameters is a non-trivial task. Existing approaches are not sufficiently accurate and do not satisfy the requirements. Thus, a load analysis and forecasting algorithm was developed that allowed to obtain more accurate estimation results and ensure the cloud computing systems congestion prevention.

To this end, a CCS load prediction model has been developed with a hybrid approach usage. Based on the proposed approaches, a hybrid algorithm for load balancing in the cloud computing system has been built, which has confirmed its effectiveness. The generalized algorithm scheme includes solving clustering and forecasting problems and contains of the following steps:

– obtaining historical values of dynamic cloud parameter x_i per day;

– performing parameter values clustering using *c*-means values. The output is a set of *C* clusters;

– artificial neural network synthesis and training for each cluster. The output is a set of *C* neural networks;

– neural network-cluster pairs selection with the lowest mean square error σ_{MSE};

– obtaining a forecast parameter x_i value using the selected neural network-cluster pair.

In the algorithm first step, historical data on the indicator values (load) is clustered using the basic algorithm of fuzzy c-means (FCM) [9]. This is necessary in order to more accurately classify the current situation and thus improve the forecast efficiency [10]. The clustering algorithm input uses load patterns extracted from historical data using the overlapping floating window method [10], the essence of which is to identify similar trends in changing parameter values in historical data. Thus, during the algorithm operation, a set of clusters is forming, united by the similarity of load [8].

In order to forecast the load, an Elman neural network is used for each resulting cluster. It has quite high flexibility due to the fact that the number of Elman network contextual neurons is determined not by the output dimension, but by the hidden neurons number, which, in turn, made it possible to adapt the Elman neural network to solve a specific practical problem [1]. At the same time, the network is configured using the artificial immunity algorithm (AIS-WElman [7]) using the available historical values in the cluster. To perform the forecast, such a neural network-cluster pair is selected, for which the forecastion error is minimized.

In this algorithm, L data points are given for each CCS host each dynamic characteristic on the timeline: $s_n, s_n \in R, n=1,2,..,L$. The end goal is to forecast the value s_{L+1}, that is, the next characteristic value in the timeline. Clustering and forecasting using the FCM algorithm and AIS-WElman networks occurs as follows:

– overlapping fixed-size sliding window is used to highlight the values sequences $s_1,..,s_k$ of the next dynamic characteristic x_i (load on the cloud *i*-th component). This reveals similar load patterns;

– based on identified patterns, N data points are created with a d-dimensional vector x_i, where $x_i=\{s_1,s_2,..,s_{i+d+1}\}\in R, n=1,2,..,N$, where $N=L-d+1$;

– Elman networks creation for each cluster by clustering N data points in C indistinct sets, i.e. member u compliance degree definition for each sequence values s of characteristic x_i, $0\le u_{ic}\le 1$ in different clusters of c, $1\le c\le C$, so $\sum_{c=1}^{C}u_{ic}=1$;

– forecasting dynamic characteristic value obtaining x_i for each cluster using an Elman network. Each Elman network is trained using an artificial immunity algorithm using data point values within each cluster;

– calculation of the forecasting mean square error σ_{MSE} for each neural network-cluster pair of the Elman network and network with the lowest mean square error selection;

– parameter s_{L+1} forecasting value calculation using the selected network;

– go to the next parameter.

Load forecasting is performed separately for each of the host dynamic parameters. The forecasting algorithm input value is a vector of historical data about this parameter i-th value (for example, the load of the first host first CPU core

$$L_{CPU}^{i}:\{L_{CPU_{t=1}}^{i},L_{CPU_{t=2}}^{i},..,L_{CPU_{t=n}}^{i}\}.$$

It is worth noting that different parameters values forecasting can be performed simultaneously.

Load clustering based on fuzzy c-means algorithm. Clustering is a method of multivariate data points collections separating into significant groups, where all group members have similar characteristics, and data points between different groups are not similar to each other. Clustering is widely used in various tasks, such as marketing research, image segmentation, biological species determination, and urban planning. Classical clustering methods, such as k-means and c-means, operate on clusters with a clear division, that is, each object belongs strictly to the one group. This property makes such clusters impractical for real applications, since many objects class attributes may be fuzzy, especially for multidimensional data analysis [3]. Fuzzy clustering is a method that allows to determine the samples similarity to each other degree and organize clusters by similarity. Thus, fuzzy clustering methods allow to build a more adequate load model in a cloud computing system.

However, most conventional fuzzy clustering techniques, such as fuzzy c-means and fuzzy k-means, can only operate on linearly separated data points in the observation space. These cloud computing system resources loadings are multidimensional, therefore have to be subjected to basic nonlinear transformation Φ. Let $X = \{x_1, x_2, .., x_n\}$ is a set of historical data size N in d-dimensional space R^d, Φ - nonlinear function of this entrance space and multidimensional space H display: $\Phi : R^d \to H, x \to \Phi(x)$.

The input data for clustering is historical data about load on each of the cloud computing system resources. The basic fuzzy c-means algorithm (KFCM) can divide a set of historical data into C clusters by minimizing the distance index using the mapping function Φ in the observing space:

$$\min J_{KFCM} = \sum_{c=1}^{C}\sum_{i=1}^{N} u_{ic}^{p} \cdot \left\| \Phi(x_i) - \Phi_{v_c} \right\|^2, \qquad (1)$$

where $$\sum\nolimits_{c=1}^{C} u_{ic}^{p} = 1, 2, .., N, \qquad (2)$$

where u_{ic}^{p} is the membership of the data point x_i in cluster c, defined in the range $[0,1]$, as well as the ic-th member of the membership matrix U, p is a number that determines the cluster fuzziness; $\left\| \Phi(x_i) - \Phi_{v_c} \right\|^2$ is the distance between x_i and centroid v_c in a given multidimensional space with the mapping function Φ. of Cluster c centroid in the mapped space is obtained by:

$$\Phi_{v_c} = \sum\nolimits_{j=1}^{N} u_{jc}^{p} \Phi(x_j) \Big/ \sum\nolimits_{j=1}^{N} u_{jc}^{p}. \qquad (3)$$

The distance $\left\| \Phi(x_i) - \Phi_{v_c} \right\|^2$ in space is calculated as:

$$\begin{gathered} \left\| \Phi(x_i) - \Phi_{v_c} \right\|^2 = \left[\Phi(x_i) - \frac{\sum_{j=1}^{N} u_{jc}^{p} \Phi(x_j)}{\sum_{j=1}^{N} u_{jc}^{p}} \right]^2 = \\ = \Phi(x_i) \cdot \Phi(x_i) - 2 \cdot \Phi(x_i) \frac{\sum_{j=1}^{N} u_{jc}^{p} \Phi(x_j)}{\sum_{j=1}^{N} u_{jc}^{p}} + \\ + \frac{\sum_{j=1}^{N} u_{jc}^{p} \Phi(x_j)}{\sum_{j=1}^{N} u_{jc}^{p}} \cdot \frac{\sum_{j=1}^{N} u_{jc}^{p} \Phi(x_j)}{\sum_{j=1}^{N} u_{jc}^{p}} = K_{ij} - \\ -2 \cdot \frac{\sum_{j=1}^{N} u_{jc}^{p} \Phi(x_j) K_{ij}}{\sum_{j=1}^{N} u_{jc}^{p}} + \frac{\sum_{m=1}^{M}\sum_{n=1}^{N} w_{mc}^{p} w_{nc}^{p} K_{mn}}{\sum_{m=1}^{M}\sum_{n=1}^{N} w_{mc}^{p} w_{nc}^{p}}, \end{gathered} \qquad (4)$$

where $K_{ij} = K(x_i, x_j)$ is the basic function. The radial base function (RBF) is used because it is proven to be the most efficient [1]. By minimizing the equation using the E-M algorithm with the restriction U, we obtain:

$$v_c = \sum\nolimits_{i=1}^{N} u_{ic}^{p} K(x_i, v_c) x_i \Big/ \sum\nolimits_{i=1}^{N} u_{ic}^{p} K(x_i, v_c), \qquad (5)$$

$$u_{ic} = \frac{\left(1/\left(1 - K(x_i, v_c)\right)\right)^{1/(p-1)}}{\sum_{j=1}^{C} \left(1/\left(1 - K(x_i, v_c)\right)\right)^{1/(p-1)}}, \qquad (6)$$

where $c = 1, 2, .., C$, $i = 1, 2, .., N$.

Thus, with the fuzzy c-means mathematical apparatus help, it is possible to obtain an intermediate result used in the subsequent forecasting process.

Cloud computing system load forecast using the Elman network. Elman neural network is a partially recurved neural network model, first proposed by Elman in 1990 [11]. It represents something between a multi-layered perseptron and a purely recurring network. Unlike the forward propagation cycle, the hidden layer and the output layer with variable weights connecting the two neighboring layers, the backward propagation cycle uses a context layer, which is sensitive to the input data history, so the connections between the context layer and the hidden layer are fixed. The Elman network dynamic characteristics are provided only by internal connections, so the network does not need to transmit the state as an input or training signal.

To solve the problem, a modified Elman network is used to CCS parameters future values forecasting [11]. Fig. 1 shows the modified network structure.

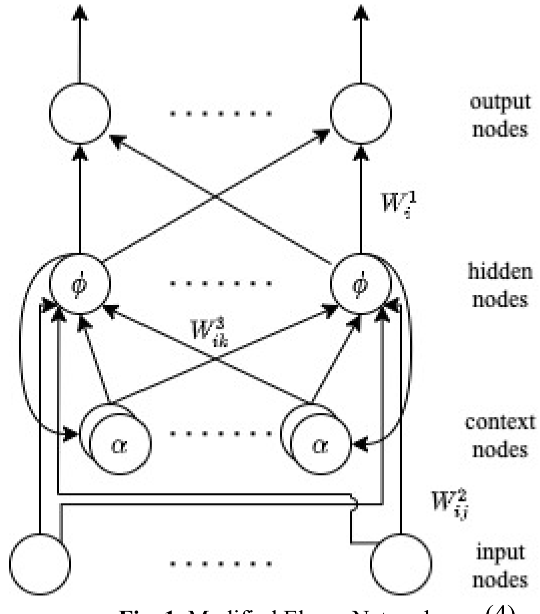

Fig. 1. Modified Elman Network (4)

The modified Elman network differs from the original one in that it has dedication connections with fixed gain α in context nodes. Thus, the context nodes output value at time t is:

$$x_{ck}(t) = \alpha x_{ck}(t-1) + H_k(t-1), \qquad (7)$$

where $x_{ck}(t-1)$ and $H_k(t-1)$ are the outputs of the k-th context nodes and k-th hidden nodes respectively, $\alpha(0 < \alpha < 1)$ is a gain in the internal connections dedication. There are n input layer nodes, m hidden and context layers nodes, as well as r output layer nodes. $x_j(t)$ are inputs at time t, and $y_j(t)$ are outputs, respectively, $W_i^1(t)$, $W_{ij}^2(t)$, $W_{ik}^3(t)$ are weights between the hidden and output layer, the input and hidden layer, the hidden and context layer respectively. The modified Elman network mathematical model the can be represented as:

$$y(t)=\sum_{i=1}^{m} W_i^1(t-1)H_i(t), \qquad (8)$$

$$H_i(t)=\varphi\big(\big(h_i(t)-b_i(t)\big)/a_i(t)\big), \qquad (9)$$

$$\begin{aligned} h_i(t)&=\sum\nolimits_{j=1}^{n} W_{ij}^2(t)x_j(t-1)+\\ &+\alpha\sum\nolimits_{k=1}^{m} W_{ik}^3(t-1)\cdot\big(x_{ck}(t-1)+H_k(t-1)\big), \end{aligned} \qquad (10)$$

where $a_i(t)$ is the expansion factor, $b_i(t)$ is the translation factor.

CCS functioning parameters forecasting neural network system generalized block diagram is shown in Fig. 2. It reflects the following basic information processing steps: CCS operation current parameters measurement; indicators historical values collection; load classification according to suitable clusters; decisive rule implementation based on the neural network.

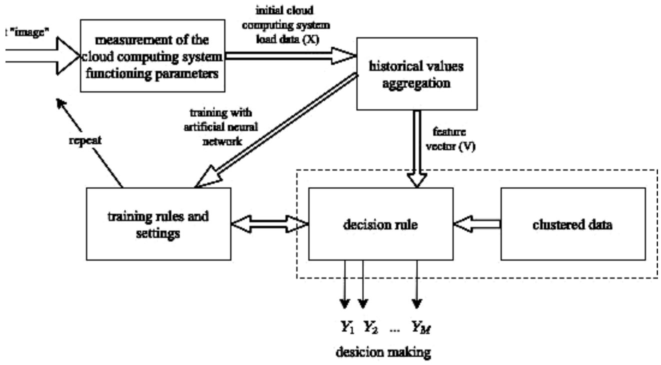
"image"
measurement of the cloud computing system functioning parameters
initial cloud computing system load data (X)
historical values aggregation
training with artificial neural network
repeat
feature vector (V)
training rules and settings
decision rule
clustered data
Y_1 Y_2 ... Y_M
desicion making

Fig. 2. CCS functioning parameters forecasting neural network system

Thus, to solve the selecting weight coefficients problem in Elman forecasting networks the artificial immune systems usage is justified. Artificial immune systems, along with genetic algorithms, are the evolutionary algorithms form; the principle of their work is similar to the mechanisms for adapting the human immune system to changing external factors. It has been proved that artificial immune systems, as a rule, are characterized by less convergence time than genetic algorithms, as well as their use significantly decreases the probability of finding locally significant solutions [1].

Solving the Elman neural network training problem with the artificial immune system help. The studies [10] of artificial neural (ANS) and immune (AIS) systems made it possible to distinguish their distinctive features, presented in Table 1.

As can be seen from the above table, AIS building concept and principles differ markedly from the ANS building concept and principles, primarily due to the greater variety (variability) of immune defense mechanisms, since the goal set for the AIS is extremely complex - to ensure the self-preservation (survival) of the observed system in an aggressive (hostile) external environment.

*Table 1 – **Distinctive features of AIS and INS***

	Artificial immune system	**Artificial neural network**
Basic elements	Lymphocytes and antibodies.	Artificial neurons.
Elements count	Isn't strictly fixed, their position changes dynamically.	Number of neurons and their location is fixed.
Network behavior	Self-organization and behavior evolution of are characteristic.	Largely depends on the accepted algorithm.
Interaction between elements	Isn't constant and can dynamically change.	Is constant and set when connected.
Mana-gement	There is no centralized management.	A single mechanism for adjusting connec-tion weights is used.

According to modern ideas, the functioning of the human immune system is based on the principle of comparing some "patterns" with bodies that have fallen into the body and detecting differences between them. The mechanisms of innate and acquired immunity are used as basic protection mechanisms. Innate immunity has little specificity and is based on the devouring and destruction of foreign bodies by macrophages and white blood cells. At this stage, only known pathogens are recognized and destroyed, the knowledge of which was acquired by the immune system in the evolution process. The acquired immunity is responsible for the recognition of specific "strangers" (molecules (antigens) not known to the system in advance), and is associated with the activity of lymphocytes - cells that play the above-mentioned patterns role. The total number of lymphocytes in the human body is about 2×10^{12}; by cell weight the human immune system is comparable to the brain. Lymphocytes are carried by the body through the lymph nodes, constantly moving and thus controlling the body as a whole, and not its individual areas. Each type of lymphocyte is responsible for detecting some limited number of antigens.

There are two main groups of lymphocytes: B-lymphocytes and T-lymphocytes. B-lymphocytes produce antibodies - special proteins from the immunoglobulins class that bind to antigens and prepare them for subsequent destruction. For successful operation, the immune system must generate a huge number of diverse antibodies capable to contact with any molecule. T-lymphocytes, like B-lymphocytes, initially originate in the spinal cord, after which they enter the thymus - an environment rich in the body's own antigens. T-lymphocytes do not produce free antibodies. There are several types of T-lymphocytes: T-helpers - cells that recognize foreign bodies, and T-killers - cells that destroy antigens identified by helpers.

An important mechanism of the immune system is negative selection. This process essence is that T-lymphocytes located in large quantities in the thymus interact with each of their own antigens, and if the T-lymphocyte "recognized" its own antigen, i.e. reacted to it as if it was a foreign cell, then lymphocyte is removed. Thus, lymphocytes "learn" not to respond to their own cells in order to recognize only foreign antigens in the future. Thus, negative selection creates "patterns" containing the information that is absent from the body, and if some body fits this pattern, it means that it is clearly "alien".

Another equally important immune response mechanism underlying immune system behavior is the clonal selection mechanism. According to this mechanism, if any of the B-lymphocytes detected a certain antigen, then the cloning process of the corresponding lymphocytes is immediately activated. The more antigens of this type enter the body, the more corresponding B-lymphocytes are formed, which produce antibodies responsible for these antigens destruction. High degree of lymphocyte and antigen affinity increases its cloning intensity and decreases lymphocyte mutations intensity. Lymphocytes with the greatest affinity degree are preserved in the form of long-lived memory cells. The life of the remaining lymphocytes is short. Thus, clonal selection provides a kind of B-lymphocytes (and their corresponding antibodies) natural selection: only those, that fit this antigen as much as possible, - survive. The AIS is based on the following natural immune systems properties:

- recognition. The immune system is able to recognize and classify different molecular structures and respond selectively to them. Recognition of one's and another's [5] is one of the main tasks solved by the immune system;
- diversity. The immune system uses a combinatorial mechanism (genetically conditioned process) to form many different lymphocyte configurations to ensure that at least one lymphocyte in the entire population can interact with any predetermined (known or unknown) antibody;
- training. The immune system evaluates the structure of a particular antigen using its random contacts with cells constituting this system. Training consists in changing the lymphocyte concentration that occurs in the primary response (as a result of the first contact with the antigen). The immune system ability to learn is mainly embedded in the mechanism of clones' replenishment, leading to the new immunocompetent cells formation, taking into account the system current state;
- memory. Using short-term and long-term immune memory mechanisms, the immune system maintains an ideal balance between resource savings and function performance by maintaining minimal but sufficient memory of previous antigen contacts;
- distributed search. In essence, the immune system is a distributed system. Immune system cells, mainly lymphocytes, are continuously recycled through the blood, lymph, lymphoid organs and other tissues. In the case of encountering an antigen, they carry out a specific immune response;
- self-regulation. The immune system has the property of self-regulation. There is no central organ that controls the immune system functions. Depending on the method of penetration into the body and other antigen properties, the immune response regulation can be both local and systemic;
- threshold mechanism. Immune response and multiplication of immunocompetent cells occur only after overcoming some threshold depending on the strength of chemical bonds;
- co-stimulation. B-lymphocyte activation is tightly controlled by an additional stimulating signal. The second signal (from helper T-lymphocytes) helps to ensure tolerance and distinguishes between a serious threat and a "false call" (i.e., dangerous and non-dangerous antigens);
- dynamic protection. The immune system not only detects and removes antigens from the body, but also participates in the body self-maintenance process in a dynamically changing environment by interactions between lymphocytes and antibodies. Thus, the immune system has a methodology for solving dynamic rather than static problems in an unknown and hostile environment. The following actions are used to build the training AIS: immune cells set generation (antibodies); generated antibodies cloning (ancestors); bowed antibodies growing (hypermutation analogue); antibodies and reference values interaction assessment (antigens); antibodies with a similarity or adaptability low value getting rid (lymphocytes).

The antibody population is initialized with binary strings representing the Elman neural network weights and displacements. Weights are calculated using the fit values. Fitness is calculated as follows:

$$fitness^{j}(k) = 1 \Big/ \left(1 + e(t)^{2}\right), \qquad (11)$$

where e is the error value during t;

$$e(t) = y(t) - d(t), \qquad (12)$$

where y is the forecast result and d is the expected result at time t.

Each antibody from the initial set is cloned several times to create a temporary clone population. The number of each antibody clones is calculated as follows:

$$N \bullet fitness^{j}(k) \Big/ \sum\nolimits_{i=1}^{N} fitness^{j}(i), \qquad (13)$$

where N is the number of antibodies of the same type, $fitness(k)$ is the value of the adaptability function of the k-th antibody relative to the j-th neuron. This clone population is designed to carry out the process of growing up through the hypermutation mechanism. Hypermutation is carried out based on the similarity degree value: the lower it is, the higher the hypermutation coefficient, and vice versa. The number of bits to be hypermuted is calculated as:

$$M \bullet \frac{\max(fitness^{j}) - fitness^{j}(k)}{\max(fitness^{j}) - \min(fitness^{i})}, \qquad (14)$$

A new antibody population of the same size as the original is then selected and the operation is repeated until the desired weight values are achieved.

Thus, an algorithm is obtained that is effective for training the Elman neural network and combines all the advantages of artificial immune systems.

Research results and discussions

Two well-established approaches are proposed to evaluate the prediction results - the analysis of mean square error (σ_{MSE}) and the mean absolute error (σ_{MAE}) [7].

$$\sigma_{MSE} = \sum_{t=1}^{h} \left(y_t - \hbar_t\right)^2 \Big/ \sum_{t=1}^{h} \left(y_t\right)^2 , \qquad (15)$$

$$\sigma_{MAE} = \sum_{t=1}^{h} \left|y_t - \hbar_t\right| \Big/ h , \qquad (16)$$

where $\hbar_t$ is the parameter forecast value, y_t is the real value, h is the number of time points whose forecast value should be calculated (in our case $h = 1$).

The training sample consisted of 300 examples, in which the number of instances varied from 2 to 1000, the number of servers from 1 to 100. The mean square error averaged for all training examples was 0.042. The average absolute error averaged for all test examples was 0.4. The mean square error and mean absolute error measurements for each forecasting method are shown in Fig. 3. The results of the error comparison with other forecasting algorithms for 15 randomly selected measurements are shown in Fig. 4. Thus, the proposed method of forecasting the change in host load in the cloud computing system allowed planning to reallocate cloud resources in order to increase their use efficiency.

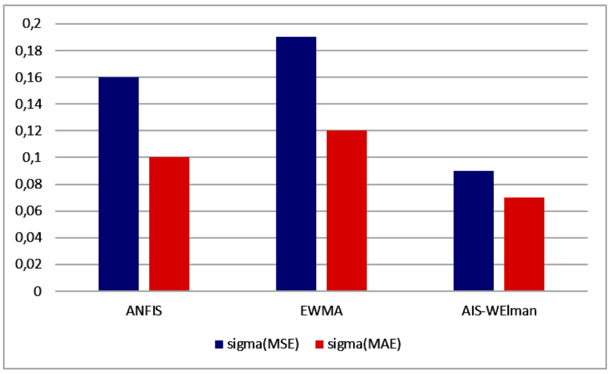

Fig. 3. Existing and developed (AIS-WElman) forecasting algorithms efficiency comparison on the mean square and mean absolute error calculation example

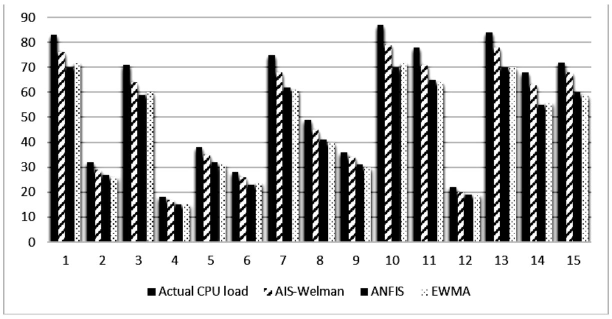

Fig. 4. Forecasting algorithms comparison results. Abscissa - CPU usage, %. Ordinate - experiment number

Conclusions and prospects for further development

1. A system analysis of existing forecasting the load on CCS resources methods has been carried out. The main differences in forecasting features, advantages and disadvantages of methods were identified. The forecasting method and algorithm based on Elman artificial neural networks selection for further improvement is justified.

2. A method for forecasting the CCS resources load has been developed, which hallmark is the use of the fuzzy c-means method for clustering historical data, as well as the use of artificial immune systems for training the neural network, which provides high accuracy in predicting the functioning parameters of a distributed computing complex taking into account the dynamics of their change. To improve forecasting efficiency, an approach based on artificial immune systems has been applied, which provides increased efficiency.

Resource load variance forecasting algorithm creation has resulted in significant improvements in cloud utilization efficiency, as well as in the ability of cloud computing systems to run new instances with minimal degradation in the already running applications performance.

The further research direction is to improve the decision-making system in order to obtain more accurate results of forecasting the CCS resources load.

References

1. Xiao, Z., Song, W. and Chen, Q. (2013), "Dynamic Resource Allocation Using Virtual Machines for Cloud Computing Environment," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 24, No. 6, pp. 1107-1117/.
2. Ashby, W. (2016), *An Introduction to Cybernetics* (in Russian), URSS, 432 p.
3. J. Vandebon, J. G. F. Coutinho, W. Luk and T. Chau (2019), "Transparent Heterogeneous Cloud Acceleration," *IEEE 30th International Conference on Application-specific Systems, Architectures and Processors (ASAP)*, New York, NY, USA, pp. 33-33, DOI: https://doi.org/10.1109/ASAP.2019.00-40.
4. Volkova, A., Shyshkunov, V. (2019), *System analysis and modeling of processes in the technosphere* (in Russian), Ural University Publishing House, Ekaterinburg, 243 p.
5. Berketov, G., Mikryukov A., Fedoseev, S. and Golovko, D. (2013), "Models and patterns recognition algorithms in life-cycle management problems of technical systems", *Innovative Information Technologies*, Vol. 3, No. 2, pp. 41-50.
6. Berketov G., Mikryukov A. and Tsurkin A. (2014), "Solving the problems of forecasting the state and managing the life cycle of complex technical complexes by methods of image recognition" (in Russian), *Statistics and Economics*, No. 1, pp. 138-143, DOI: https://doi.org/10.21686/2500-3925-2014-1-138-143.
7. Xu, D., Yang, Sh., and Luo, H. (2013), "A Fusion Model for CPU Load Prediction in Cloud Computing", *Journal of Networks*, Vol. 8, No. 11, 2506-2511, DOI: https://doi.org/10.4304/jnw.8.11.2506-2511.
8. Ramezani, F., Lu, J., Hussain, F. (2013), "A Fuzzy Predictable Load Balancing Approach in Cloud Computing", *Proceedings of the International Conference on Grid Computing and Applications (GCA)*, p. 108.
9. Bey, K.B., Benhammadi, F., Sebbak, F. (2013), "Fuzzy Subtractive Clustering Based Prediction Approach for CPU Load Availability", *The Fourth International Conference on Cloud Computing, GRIDs, and Virtualization*.
10. Saranya.S, Murugan. B.S (2014), "Intelligent Scheduling System for Dynamic Resource Allocation in Cloud Computing", *International Journal of Advanced Research in Computer Science & Technology*, Vol. 2, Issue Special 1, pp. 284-288.

11. Elman, J. (1990), "Finding structure in time", *Cognitive Science*, Vol. 14, No. 2, pp. 179-211.
12. Hrebeniuk, D. (2018), "Analysis of methods of distribution of resources in the virtualization media", *Control, Navigation and Communication Systems*, No. 6(52), pp. 98-103, DOI: https://doi.org/10.26906/SUNZ.2018.6.098.

Received (Надійшла) 14.09.2020
Accepted for publication (Прийнята до друку) 21.10.2020

ВІДОМОСТІ ПРО АВТОРІВ/ ABOUT THE AUTHORS

Давидов Вячеслав Вадимович – кандидат технічних наук, доцент кафедри обчислювальної техніки та програмування, Національний технічний університет «Харківський політехнічний інститут», Харків, Україна;
Viacheslav Davydov – Candidate of Technical Sciences, Associate Professor of Computer Engineering and Programming Department, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine;
e-mail: vyacheslav.v.davydov@gmail.com; ORCID ID: https://orcid.org/0000-0002-2976-8422.

Гребенюк Дарина Сергіївна– аспірантка кафедри обчислювальної техніки та програмування, Національний технічний університет «Харківський політехнічний інститут», Харків, Україна;
Daryna Hrebeniuk – graduate student of Computer Engineering and Programming Department, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine;
e-mail: darina.gg1@gmail.com; ORCID ID: https://orcid.org/0000-0001-5331-2444.

Розробка методу прогнозування зміни навантаження ресурсів в системах хмарних обчислень

В. В. Давидов, Д. С. Гребенюк

Анотація. Предметом дослідження в статті є моделі і методи прогнозування навантаження в умовах хмарних обчислень з використанням математичного апарату нейронних мереж. **Метою** роботи є підвищення ефективності використання ресурсів системах хмарних обчислень (таких, як оперативна пам'ять, дисковий простір, ЦПУ, мережа) шляхом розробки методів прогнозування використання навантаження ресурсів. У статті вирішуються такі **задачі**: розробка комплексного підходу до завдань прогнозування навантаження ресурсів хмарних систем, що включає в себе синтез комбінованої прогнозуючої нейронної мережі; розробка прогнозної нейросетевої моделі на базі нейронної мережі Елмана; розробка методу навчання нейронної мережі на основі алгоритму штучного імунітету; оцінка ефективності розробленого методу. Для вирішення поставлених завдань були використані підходи і **методи** штучних нейронних та імунних систем, а також методи теоретичних досліджень, які засновані на наукових положеннях теорії штучного інтелекту, статистичного, функціонального і системного аналізів. Отримані наступні **результати**: на основі проведеного аналізу методів прогнозування різних навантажень в системах систем обчислень були виявлені основні особливості існуючих методів, продемонстровані їх переваги і недоліки. На основі проведеного аналізу результатів дослідження доведено доцільність вдосконалення аналітичних методів прогнозування навантаження. Удосконалено метод прогнозування навантаження обчислювальних ресурсів в системах хмарних обчислень, що дозволяє отримати більш точні результати оцінки і забезпечити перевантажень в системах хмарних обчислень. Отримані результати підтверджені проведеними експериментами при використанні коштів інфраструктури приватних інфраструктурних сервісів. **Висновки**: удосконалення методу прогнозування навантаження на базу математичного апарату штучних нейронних мереж. Забезпечити планування розмежування ресурсів з метою підвищення ефективності їх використання.

Ключові слова: нейронна мережа Елмана; системи хмарних обчислень; прогнозування визначення навантаження.

Разработка метода прогнозирования варьирования нагрузки ресурсов в системах облачных вычислений

В. В. Давыдов, Д. С. Гребенюк

Аннотация. Предметом исследования в статье являются модели и методы прогнозирования варьирования нагрузки в системе облачных вычислений с использованием математического аппарата нейронных сетей. **Целью** работы является повышение эффективности использования имеющихся ресурсов в системах облачных вычислений (таких, как оперативная память, дисковое пространство, ЦПУ, сеть) путем разработки метода прогнозирования варьирования нагрузки. В статье решаются следующие **задачи**: разработка комплексного подхода к прогнозированию варьирования нагрузки ресурсов облачных системах, которая включает в себя синтез комбинирующей прогнозирующей нейронной сети; разработка прогнозной нейросетевой модели на базе нейронной сети Элмана; разработка метода обучения нейронной сети на основе алгоритма искусственного иммунитета; оценка эффективности разработанного метода. Для решения поставленных задач были использованы подходы и **методы** искусственных нейронных и иммунных систем, а также методы теоретических исследований, которые основаны на научных положениях теории искусственного интеллекта, статистического, функционального и системного анализов. Получены следующие **результаты**: на основе проведенного анализа существующих методов прогнозирования варьирования нагрузки ресурсов в системах облачных вычислений были выявлены основные особенности существующих методов, приведены их достоинства и недостатки. На основе проведенного аналитического исследования доказана необходимость совершенствования существующих методов прогнозирования нагрузки. Усовершенствован метод прогнозирования нагрузки вычислительных ресурсов в системах облачных вычислений, позволяющий получить более точные результаты оценки и обеспечить предотвращение перегрузок в системах облачных вычислений. Полученные результаты подтверждены проведенными экспериментами при использовании программного обеспечения для создания частных инфраструктурных облачных сервисов и облачных хранилищ. **Выводы**: усовершенствование метода прогнозирования варьирования нагрузки на базе математического аппарата искусственных нейронных сетей Элмана в системах облачных вычислений позволило обеспечить планирование разграничения облачных ресурсов с целью повышения эффективности их использования.

Ключевые слова: нейронная сеть Элмана; системы облачных вычислений; прогнозирование варьирования нагрузки.