generalization, and also reducing the structural and parametric complexity of models synthesized on their basis.

In the works [13–16] proposed methods of extracting production rules based on decision trees [13], sets of associations [14] and negative selection [15]. The disadvantage of such methods is their serial implementation, which makes it difficult to use them when solving practical problems of large dimension. To eliminate this drawback, a parallel method for production rules extraction based on intelligent calculations has been developed, using the methods proposed in [13–16] as a basis. It provides the parallelization of the most resource-intensive operations of these methods.

For practical use of the parallel method of production rules extraction, it is necessary to use high-performance computer systems, such as clusters and graphics processors. At the same time, in order to effectively use them, it is necessary at first to estimate the used resources and predict the outcome indicators [17–19]. This allows rational use of expensive computer systems to obtain the expected result.

The purpose of the work is a construction of the model for estimation parallel computer systems resources used to solve applied problems based on the parallel method of production rules extraction.

1 PROBLEM STATEMENT

While estimating the amount of parallel system resources used, the important characteristics that determine effectiveness of the system application (the speed of obtaining result t and amount of used RAM Mem) are the following parameter groups: the technical characteristics of a parallel system, the parameters of used (mathematical) software and the characteristics of the applied problem [13–16].

The main characteristics of the system that affect the time and amount of used RAM of solving the practical problem are:

– x_1 is a type of system $type$ (cluster of CPU or GPU);

– x_2 is a number of processes on which problem is executed N_{pr};

– x_3 is a network bandwidth V, Gb/s.

The main parameters of used mathematical software (in this case, parallel method of production rules extraction [13–16]):

– x_4 is number of parallel system nodes α_{DT}, involved for production rules extraction based on decision trees [13];

– x_5 is portion of parallel system nodes α_{AR}, involved for knowledge extraction based on associative rules [14];

– x_6 is portion of parallel system nodes α_{NS}, involved for production rules extraction based on negative selection [15, 16].

As parameters of an applied problem significantly affect the amount of used RAM and the speed of work of a parallel system for parallel method of production rules extraction, we can use:

– x_7 is a number of cases Q in a given training sample observations S;

– x_8 is a number of features M in a variety of observations S.

Thus, to estimate the amount of used recourses of parallel computer system Vol (operating time (t) and amount of used RAM Mem) while executing the method of production rules extraction, need to build a model of the form (1):

$$Vol = \{t, Mem\} = Vol\left(type, N_{pr}, V, \alpha_{DT}, \alpha_{AR}, \alpha_{NS}, Q, M\right), \quad (1)$$

allowing to execute the prediction of time spent and amount of RAM used to perform the parallel method of production rules extraction depending on the characteristics of the system, the parameters of software and features of solved applied problem.

2 LITERATURE REVIEW

In the works [13–16] methods of extracting knowledge in the form of production rules based on decision trees, sets of associations and negative selection.

Stochastic method of extracting rules based on decision trees [13] uses the information about informativeness of features, difficulty of synthesized tree, as well as the accuracy of his recognition. This allows at the initial stage to form a set of tree structures which characterized by a simple hierarchy and a low recognition error, in the search process to create new sets of solutions, taking into account information about the significance of features and the interpretability of generated trees, which in turn provides the ability to build a decision tree with a small amount elements (nodes and links between them) and acceptable accuracy of recognition, as well as the extraction on its basis the most valuable instances.

The method of numerical associative rules extraction [14] involves preliminary splitting of values of the features into intervals (terms), taking into account the width of range of values and frequency of features falling into each of terms, uses the stochastic approach for searching various combinations of antecedents and consequents of associative rules, uses apriori information about the significance of the terms and features, that allows to process numerical information while extracting association rules, do not realize a substantial number of passages on the given base of transactions, identify rules with a high level of reliability and other criteria evaluation of their quality.

The proposed method of the synthesis of production rules on the basis of negative selection [15] for the case of the uneven distribution of instances of sample classes uses known information about instances of sample classes in the generation of a set of detectors, takes into account information about the individual significance of features, as a form of detector uses a hypercube maximum possible volume, allowing to exclude irrelevant and redundant features from the sample, thereby reducing the search space and run time of the method, as well as to form a set of detectors with high approximating and generalizing abilities.

However, the proposed in [13–16] methods based on processing data in serial mode, that limits their practical use because of presence of practical dimension threshold limit of data processed. In order to enable the processing of high dimensional data, a parallel method of production rules extraction has been developed. It is based on [13–16] serial methods. So, it is proposed to perform in a parallel mode the most resource-intensive operations of methods associated with calculating the values of the objective function, creating new sets of solutions, etc.

For practical application of the proposed method of production rules extraction based on intelligent calculations, it is necessary to develop the mathematical and computer software for estimation the amount of used resources of a parallel computer system.

3 MATERIALS AND METHODS

The proposed method [13–16] has been applied in a cluster of CPU and in the GPU at data processing from a public repository [20], as well as for solving practical problems [21–23]. Characteristics of processed data sets are shown in Table 1.

As a result of data processing [20–23] a training sample (2) has been formed. It has contained 1032 results of method execution, each of which has characterized by eight characteristics:

$$D = \langle X, Vol \rangle, \qquad (2)$$

where $X=\{x_1,x_2,x_3,x_4,x_5,x_6,x_7,x_8\}, x_i = \{x_{i1},x_{i2},\dots,x_{iN}\}, N=1032$, $Vol = \{t, Mem\} = \{\{t_1,t_2,...,t_N\},\{mem_1, mem_2 ..., mem_N\}\}$.

Consequently, the training sample has been formed as the table of numbers consisting of 1032 rows and 10 columns containing values of 8 inputs and 2 outputs attributes (system's time spent and an amount of RAM used) for each case of use of this method in a parallel system. Fragment of the training sample is given in Table 2.

To eliminate the influence of different value orders of features (attributes) to the synthesized model the normalization of attributes has been performed, i.e. bringing their range of feature values to the single interval [0;1].

As a basis for model (1) construction the feed forward neural network has been used, allowing approximate complex nonlinear dependencies with high accuracy. Model (1) was synthesized in the form of a three-layer perceptron [24, 25]. The first layer has contained four neurons, the second layer – four neurons, the third layer – two neurons (according to the number of model outputs). All neurons have had sigmoid activation function $\psi(\varphi)$. As a discriminant function of neurons a weighted sum has been used [25].

Table 1 – Numerical characteristics of tasks for parallel method of production rules extraction

Task					
№	Name	*M*	*Q*	Features	Output parameter
1	Recognition of motor vehicles [21]	16384	10000	integer	binary
2	Diagnosing the quality of life of patients with chronic obstructive bronchitis on the combination of used medicines [22]	95	1023	binary	binary
3	Diagnosing gas turbine aircraft engine blades by the spectra of free damped oscillations [22]	10240	318	real	binary
4	Recognition of the plant types by the spectral points [22]	55	248	real	binary
5	Construction of confectionery quality model [23]	43	956	substantial (data with gaps)	real
6	Communities and Crime [20]	128	1994	real	real
7	Parkinsons Telemonitoring [20]	26	5875	real	integer
8	Energy efficiency [20]	8	768	real	real
9	Concrete Compressive Strength [20]	9	1030	real	real
10	Forest Fires [20]	13	517	real	real

Table 2 – Fragment of the training sample

Feature values								*Vol*	
type	N_{pr}	*V*	α_{DT}	α_{AR}	α_{NS}	*Q*	*M*	*T*	*Mem*
0	1	20	0.2	0.3	0.5	318	10240	216.01	25.34
0	10	20	0.2	0.3	0.5	318	10240	40.98	290.67
…	…	…	…	…	…	…	…	…	…
0	3	1	0.25	0.40	0.35	318	10240	107.84	83.48
0	7	1	0.25	0.40	0.35	318	10240	52.38	198.25
…	…	…	…	…	…	…	…	…	…
0	12	20	0.1	0.30	0.6	318	10240	34.15	354.77
0	21	20	0.1	0.30	0.6	318	10240	25.47	521.72
…	…	…	…	…	…	…	…	…	…
1	140	32	0.2	0.3	0.5	318	10240	69.99	27.33
1	240	32	0.2	0.3	0.5	318	10240	40.83	24.85
…	…	…	…	…	…	…	…	…	…
1	60	32	0.27	0.3	0.43	956	43	2.56	0.35
1	180	32	0.27	0.3	0.43	956	43	0.85	0.33
…	…	…	…	…	…	…	…	…	…
0	26	20	0.2	0.3	0.5	1023	95	0.78	22.75
0	32	20	0.2	0.3	0.5	1023	95	0.70	28.47
…	…	…	…	…	…	…	…	…	…

Thus, a structure of three-layer synthesized neural model T_{NN} can be represented as follows (3):

$$
\begin{aligned}
&Vol=\left\{\psi_{(3,1)}\left(\varphi_{(3,1)}\left(w_{(3,1)};\psi_{(2)}\right)\right),\psi_{(3,2)}\left(\varphi_{(3,2)}\left(w_{(3,2)};\psi_{(2)}\right)\right)\right\};\\
&Vol=\{T,Mem\};\\
&\psi_{(2)}=\left\{\psi_{(2,1)},\psi_{(2,2)},\psi_{(2,3)},\psi_{(2,4)}\right\};\\
&\psi_{(2,k)}=\psi_{(2,k)}\left(\varphi_{(2,k)}\left(w_{(2,k)};\psi_{(1)}\right)\right),\ k=1,2,3,4;\\
&\psi_{(1)}=\left\{\psi_{(1,1)};\psi_{(1,2)};\psi_{(1,3)};\psi_{(1,4)}\right\};\\
&\psi_{(1,l)}=\psi_{(1,l)}\left(\varphi_{(1,l)}\left(w_{(1,l)};X\right)\right),\ l=1,2,3,4.
\end{aligned}
\tag{3}
$$

For neural model synthesis and determining of its parameters (weights and biases of each neuron) normalized features have been fed into it inputs. The outputs are the values of parallel computer system's time spent and RAM used.

The minimum of mean square error *MSE* has been used as the objective function of neural model training. The acceptable level of mean square error 10^{-4} has been considered.

After substituting of the obtained weighs and biases of neural network to the (3) and using the activation function and the discriminant function, the mathematical description of the synthesized neural network model (4) has been obtained. It describes the relationship between the characteristics of a parallel system, in which the method of production rules extraction is performed, the parameters of the investigated method, and the time spent and the amount of RAM used. Graphic interpretation of synthetic model is shown in Fig. 1. The value of mean square error of the synthesized model is $1.92\cdot10^{-4}$, which is acceptable for this sort of problems.

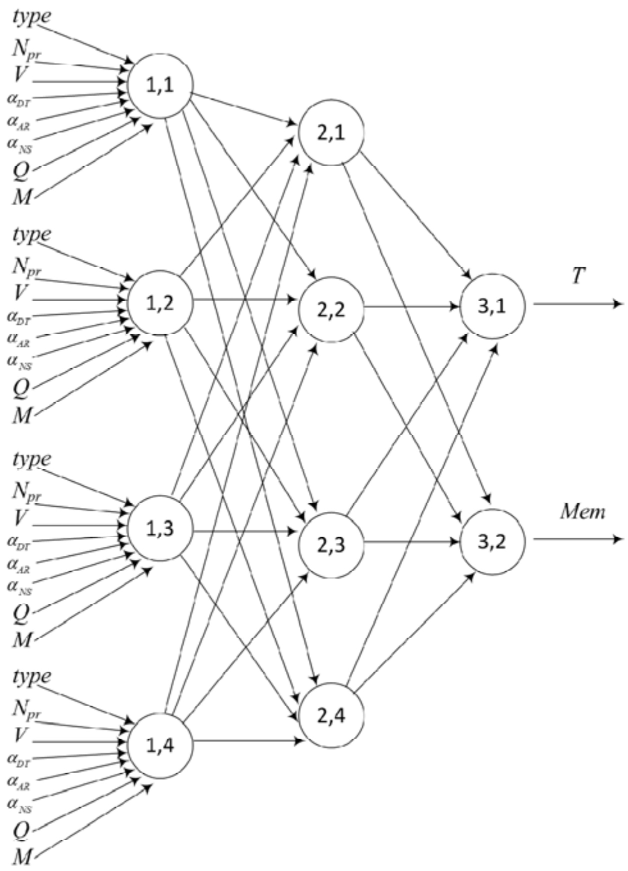

Figure 1 – Synthesized neural network model

$$
\begin{aligned}
&T=\psi_{(3,1)}=\ 1+e^{-\left(3.8484-0.0145\psi_{(2,1)}-0.0481\psi_{(2,2)}+3.9240\psi_{(2,3)}+0.1388\psi_{(2,4)}\right)}{}^{-1};\\
&Mem=\psi_{(3,2)}=\ 1+e^{-\left(-0.8644+1.2433\psi_{(2,1)}-1.2925\psi_{(2,2)}-0.6666\psi_{(2,3)}+0.3006\psi_{(2,4)}\right)}{}^{-1};\\
&\psi_{(2,1)}=\ 1+e^{-\left(-1.6567+3.0591\psi_{(1,1)}-0.2137\psi_{(1,2)}+0.2245\psi_{(1,3)}-0.1405\psi_{(1,4)}\right)}{}^{-1};\\
&\psi_{(2,2)}=\ 1+e^{-\left(-4.8177+4.9222\psi_{(1,1)}-1.3346\psi_{(1,2)}+1.6105\psi_{(1,3)}-1.5260\psi_{(1,4)}\right)}{}^{-1};\\
&\psi_{(2,3)}=\ 1+e^{-\left(-0.6787-6.9746\psi_{(1,1)}-0.6947_{(1,2)}+0.7786\psi_{(1,3)}+5.1166\psi_{(1,4)}\right)}{}^{-1};\\
&\psi_{(2,4)}=\ 1+e^{-\left(2.5299+1.0410\psi_{(1,1)}+0.6746\psi_{(1,2)}+0.2966\psi_{(1,3)}-1.4787\psi_{(1,4)}\right)}{}^{-1};\\
&\psi_{(1,1)}=\ 1+e^{-\left(2.6135+0.3796type+6.4181N_{pr}+0.0043V+0.5048\alpha_{DT}+0.1782\alpha_{AR}+0.5110\alpha_{NS}-0.2267Q-3.2789M\right)}{}^{-1};\\
&\psi_{(1,2)}=\ 1+e^{-\left(-0.4403+-2.1023type-0.2210N_{pr}+0.0748V-0.4068\alpha_{DT}-2.0229\alpha_{AR}+2.8968\alpha_{NS}+0.2301Q-0.2039M\right)}{}^{-1};\\
&\psi_{(1,3)}=\ 1+e^{-\left(-0.3057+1.6824type-0.4436N_{pr}+0.7853V-1.0850\alpha_{DT}-0.3828\alpha_{AR}+2.6398\alpha_{NS}-0.0768Q-0.7402M\right)}{}^{-1};\\
&\psi_{(1,4)}=\ 1+e^{-\left(-1.5641+2.3682type-0.9614N_{pr}+0.0142V-0.8409\alpha_{DT}-0.444\alpha_{AR}-1.0934\alpha_{NS}+0.2893Q+2.4426M\right)}{}^{-1}.
\end{aligned}
\tag{4}
$$

Thus, the constructed neural network model is a hierarchical polyalgorithmic structure containing neural-like computational elements and allows estimating resources used by a parallel computer system for production rules extraction while modeling complex objects and processes.

4 EXPERIMENTS

To provide the experimental researches of the proposed neural network model the following computer systems have been involved:

– the cluster of Pukhov Institute for Modeling in Energy Engineering NAS of Ukraine (IPME, c. Kyiv) : processors Intel Xeon 5405, RAM – 4×2 GB DDR-2 for each node, communication environment InfiniBand 20Gb/s, middleware Torque and OMPI;

– GPU NVIDIA GTX 960 1024 CUDA nodes.

During experiments the number of processes involved x_2 varied from 1 to 32 for clusters and from 60 to 260 for GPUs. Network bandwidth x_3 – from 1 to 20 Gb/s, portions α_{DT}, α_{AR}, α_{NS} – from 0 to 1 (at that $\alpha_{DT} + \alpha_{AR} + \alpha_{NS} = 1$), number of observations Q and number of features M – according to Table 1. To perform the experiments the software, based on C language with the MPI library [26] and CUDA [27], have been developed.

5 RESULTS

The results of the experiments in cluster with M=10240; Q=318; $\alpha_{DT} = 0.2$; $\alpha_{AR} = 0.3$; $\alpha_{NS} = 0.5$ are shown in fig. 2 and 3. Solid line demonstrates the actual time spent and memory consumption during the execution of parallel method of production rules extraction [13–16]. Additionally dotted line displays predicted time spent and memory consumption while using the proposed model.

The results of the experiments in cluster with M=1023; Q=95; $\alpha_{DT} = 0.2$; $\alpha_{AR} = 0.3$; $\alpha_{NS} = 0.5$ are shown in fig.4 and fig. 5.

The results of the experiments in graphical processor with M=10240; Q=318; $\alpha_{DT} = 0.15$; $\alpha_{AR} = 0.28$; $\alpha_{NS} = 0.57$ are demonstrated in fig. 6 and 7.

The results of the experiments in GPU with M=1023; Q=95; $\alpha_{DT} = 0.2$; $\alpha_{AR} = 0.3$; $\alpha_{NS} = 0.5$ are displayed in fig.8 and fig. 9.

6 DISCUSSION

The test sample has consisted of 149 experiment results and included ones of practical tasks solutions in a parallel system, which have not been included into training set.

Figures 2–8 show task solution time and amount of used memory which have been calculated with the proposed model, are a little bit smaller compared to the actual time spent and the amount of memory used. This is due to a significant variety of memory and time spent while synchronizations and transfers of data between the cluster processors or GPU threads. At the same time, the more processes or threads has been involved, the more impact of synchronizations and transfers and the greater the deviation between actual and predicted time and memory.

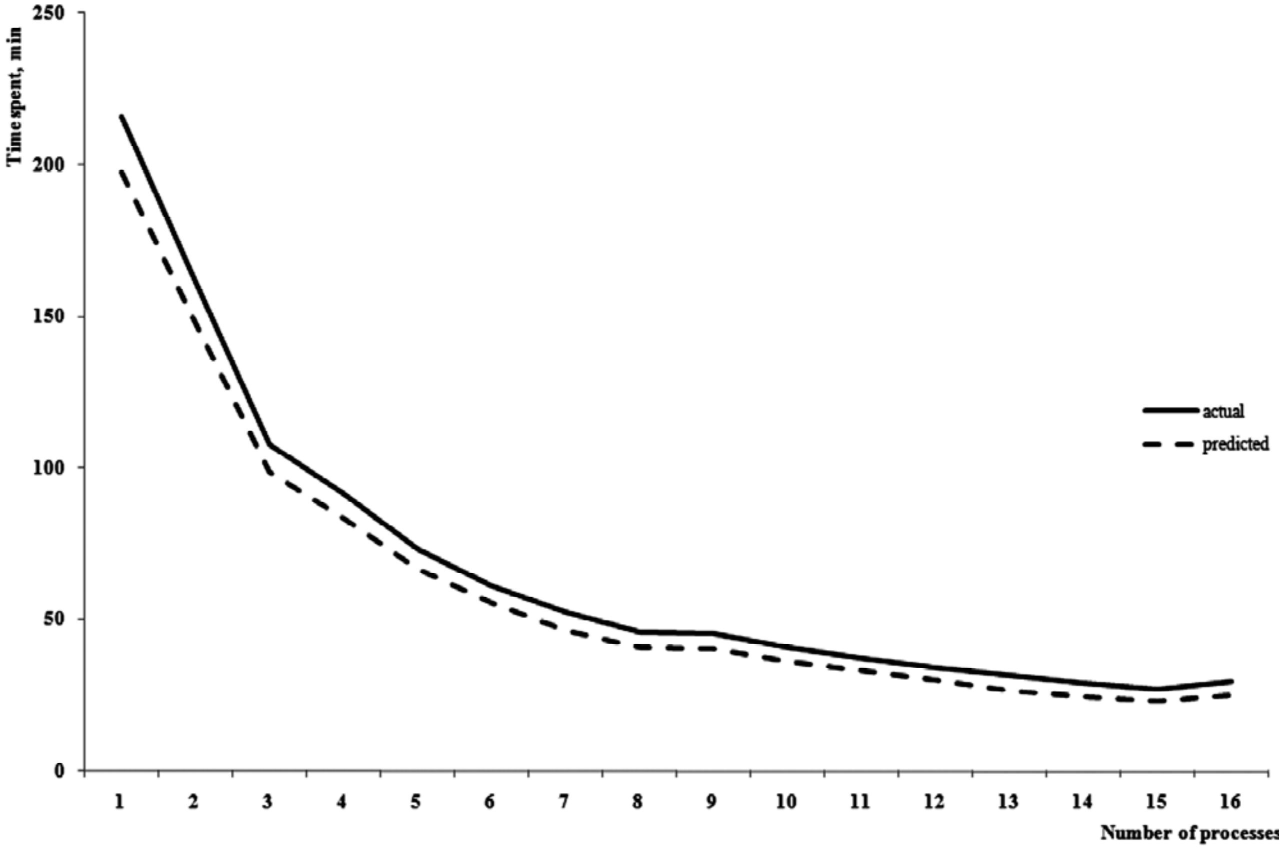

Figure 2 – Actual and predicted time spent in cluster

(M=10240; Q=318; $\alpha_{DT} = 0.2$; $\alpha_{AR} = 0.3$; $\alpha_{NS} = 0.5$)

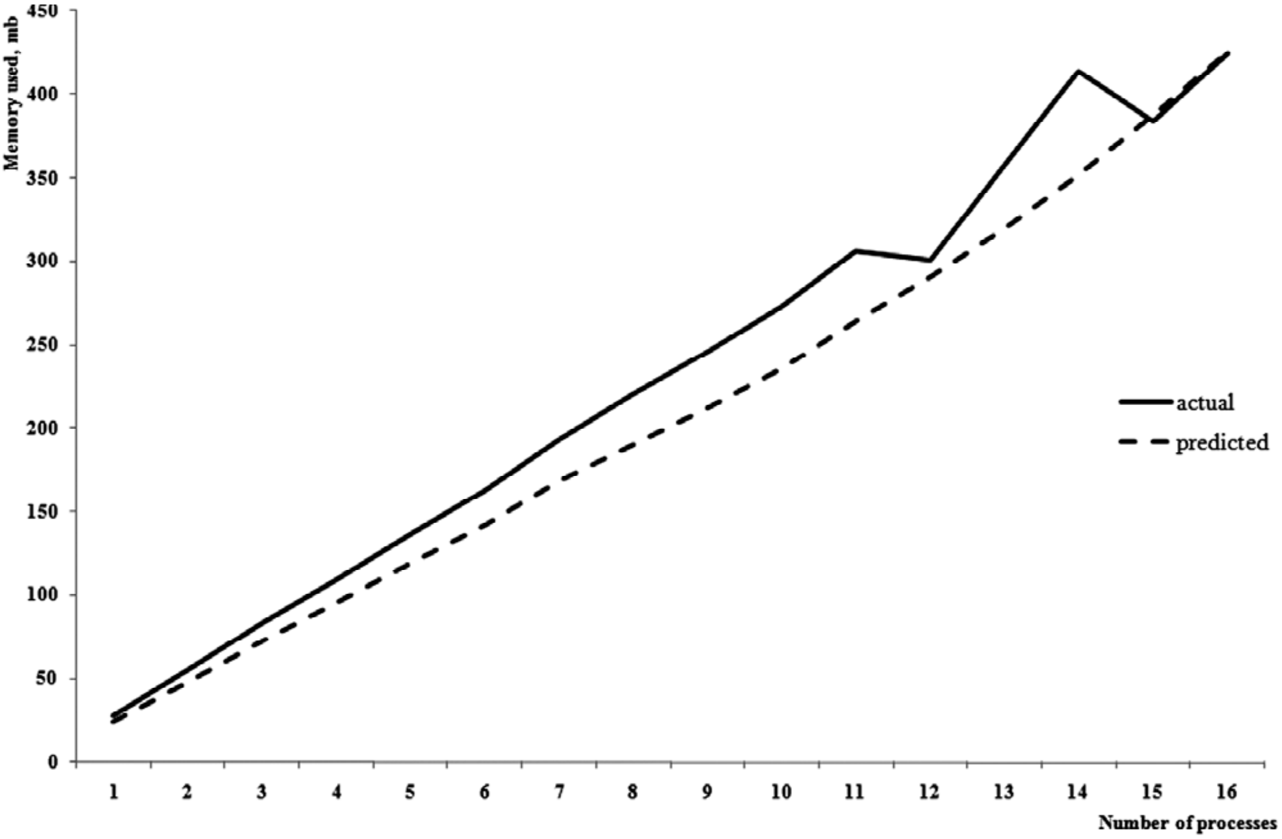

Figure 3 – Actual and predicted amount of memory consumption in cluster
(*M*=10240; *Q*=318; $\alpha_{DT} = 0.2$; $\alpha_{AR} = 0.3; \alpha_{NS} = 0.5$)

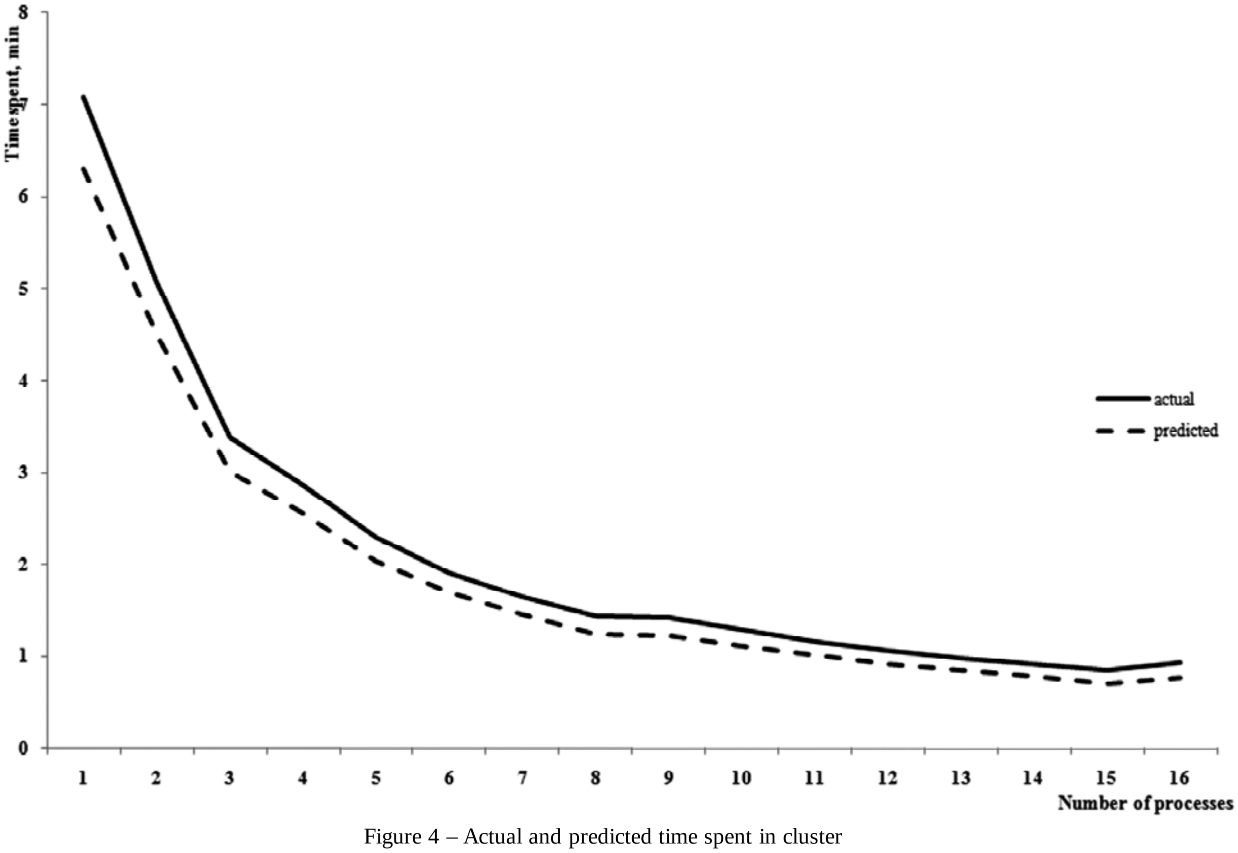

Figure 4 – Actual and predicted time spent in cluster
(*M*=1023; *Q*=95; $\alpha_{DT} = 0.2$; $\alpha_{AR} = 0.3; \alpha_{NS} = 0.5$)

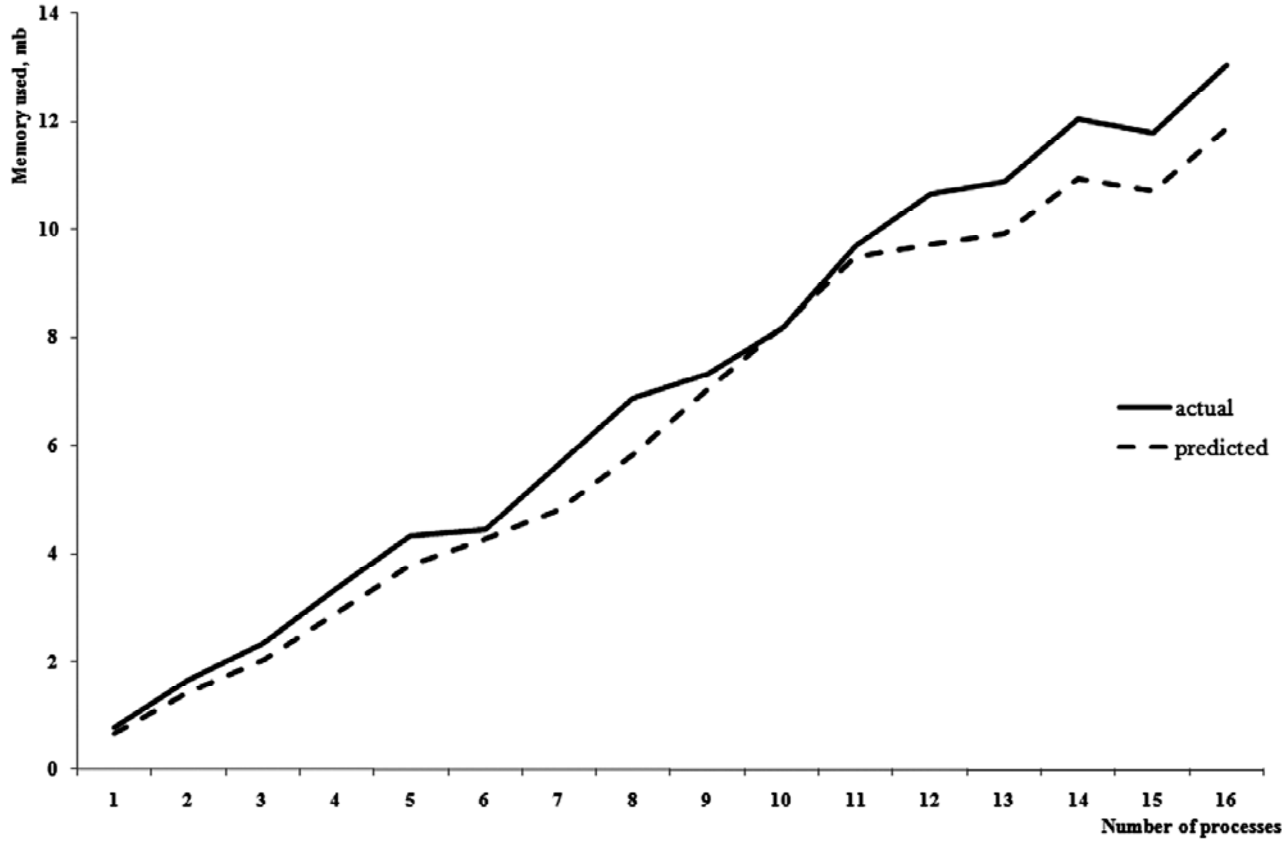

Figure 5 – Actual and predicted amount of memory consumption in cluster

(*M*=1023; *Q*=95; $\alpha_{DT} = 0.2$; $\alpha_{AR} = 0.3$; $\alpha_{NS} = 0.5$)

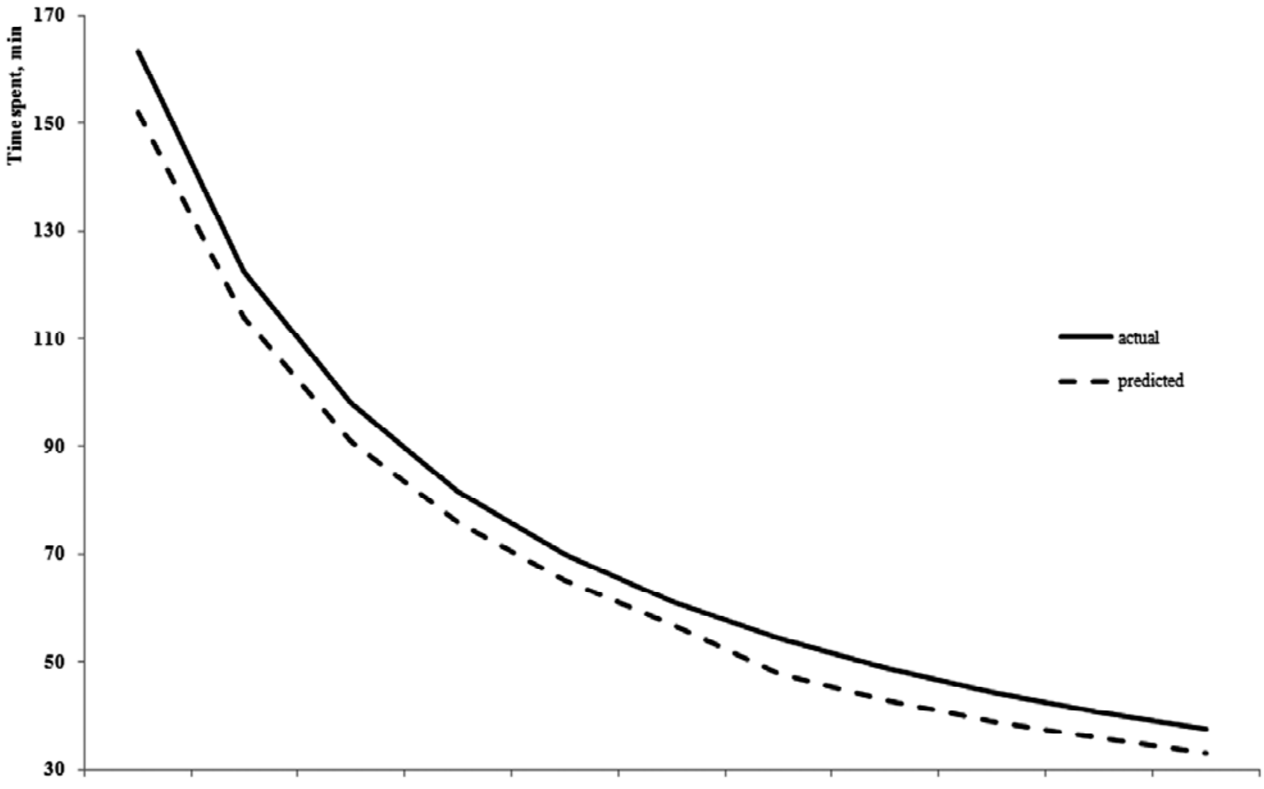

Figure 6 – Actual and predicted time spent in GPU

(*M*=10240; *Q*=318; $\alpha_{DT} = 0.15$; $\alpha_{AR} = 0.28$; $\alpha_{NS} = 0.57$)

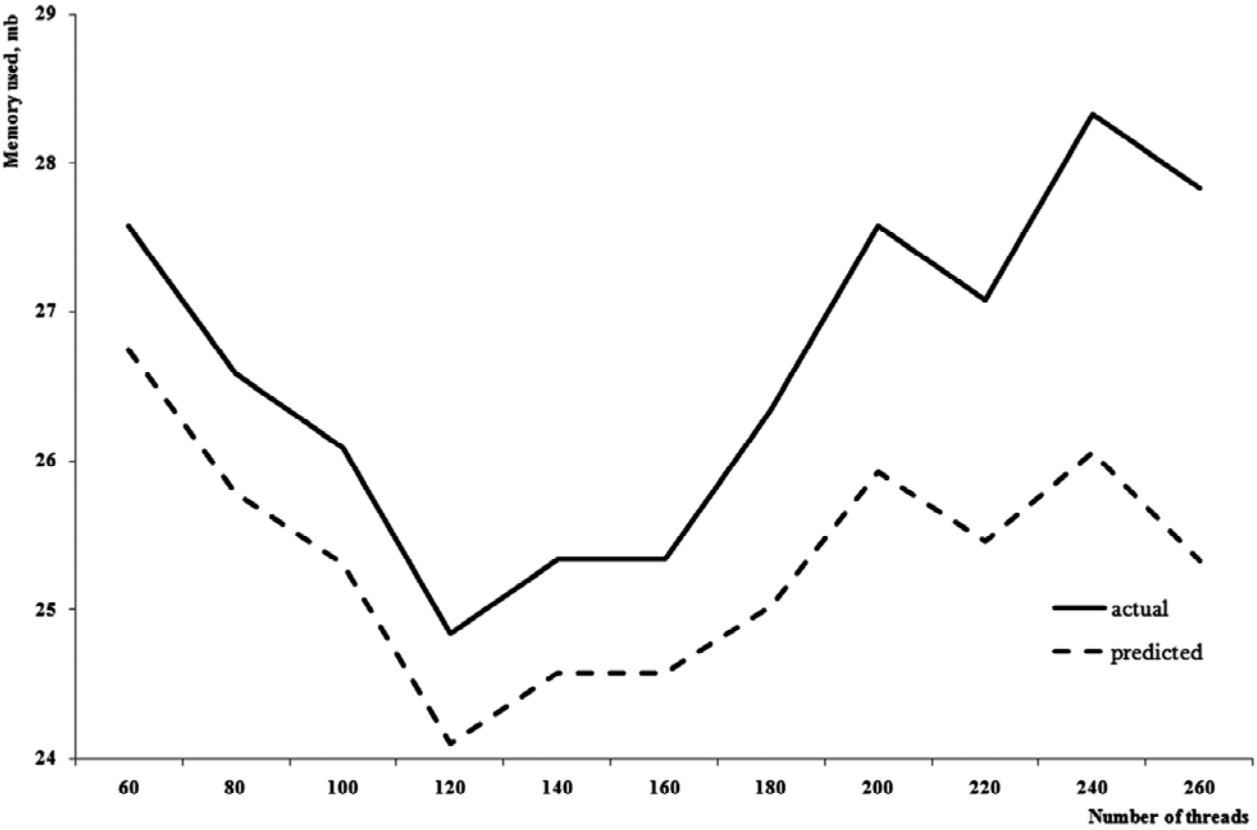

Figure 7 – Actual and predicted amount of memory consumption in GPU
(*M*=10240; *Q*=318; $\alpha_{DT} = 0.15$; $\alpha_{AR} = 0.28$; $\alpha_{NS} = 0.57$)

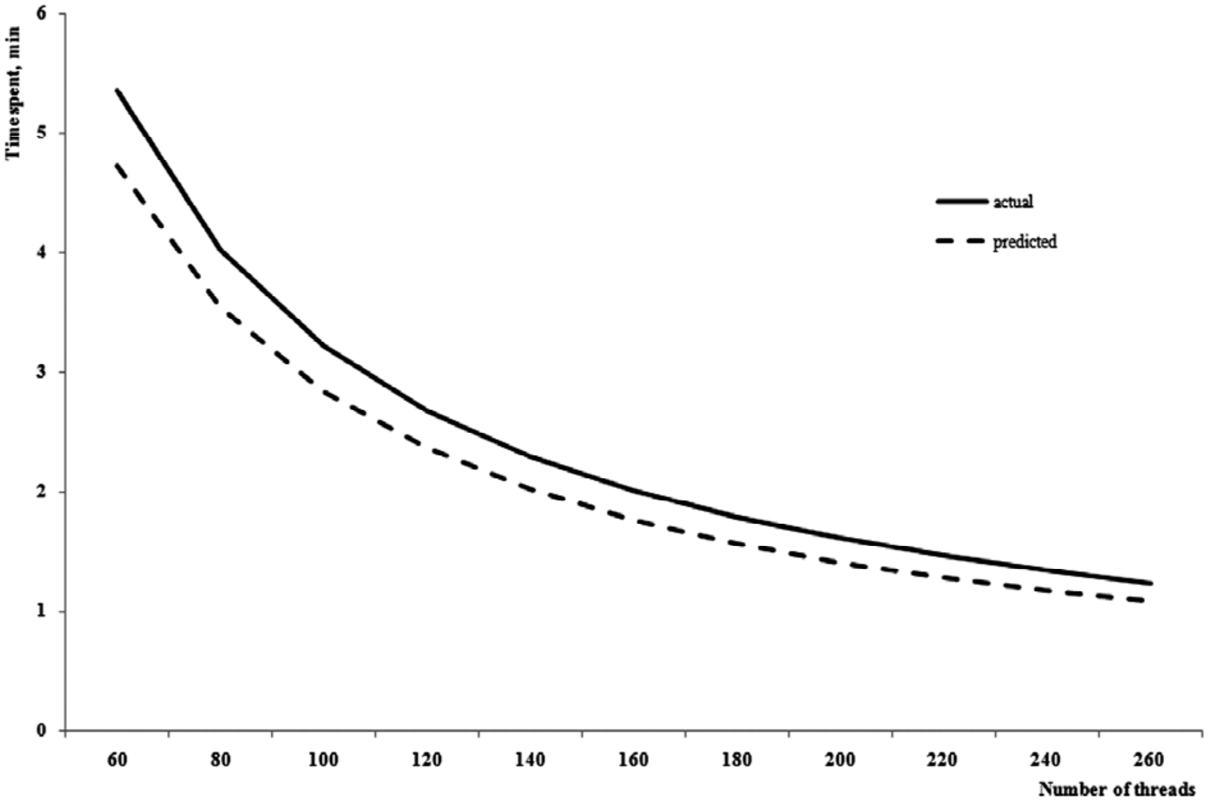

Figure 8 – Actual and predicted time spent in GPU
(*M*=1023; *Q*=95; $\alpha_{DT} = 0.2$; $\alpha_{AR} = 0.3$; $\alpha_{NS} = 0.5$)

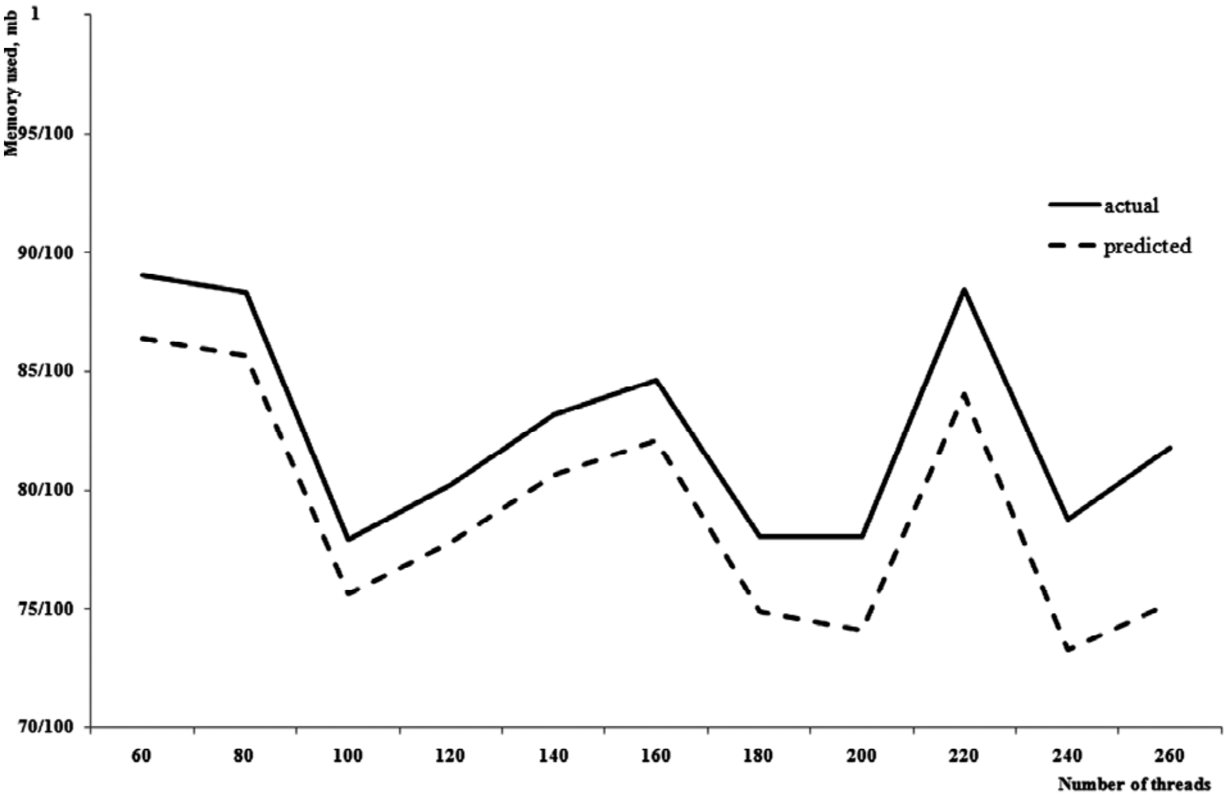

Figure 9 – Actual and predicted amount of memory consumption in GPU (M=1023; Q=95; $\alpha_{DT} = 0.2$; $\alpha_{AR} = 0.3$; $\alpha_{NS} = 0.5$)

Figures 3, 5, 7, 9 give evidence of the fact that GPU uses significantly less RAM in compare to cluster of CPUs while execution the parallel method of production rules extraction. This is because of data sample has been copied to a global GPU memory which is used by all threads. So there is no necessity to transfer the data to a local memory of each thread. Whereas each CPU in a cluster has its own memory pool which is inaccessible to the other CPUs, hence, each CPU has to copy a data sample to it's memory pool.

Mean square error MSE=$2.31 \cdot 10^{-3}$ calculated for the amount of used RAM *Mem* is much lower than MSE=$9.64 \cdot 10^{-2}$ calculated for time spent T. This follows from the fact that there are fewer factors which are affecting to the memory consumption process during the execution of the discussed method. Also memory consumption is a better predictable than the computational process.

Generally, both *MSE* values are in acceptable range, which allows recommending using the proposed model in practice.

CONCLUSIONS

The actual task of used resources estimation of parallel computer system while extracting production rules has been solved.

Scientific novelty lies in the fact that a model for estimating the resources of a computer system used when performing a parallel method of production rules extraction has been proposed. The model takes into account the type of computer system, the amount of processors involved to solving the task and the bandwidth of data transfer network. Also the model considers parameters of used mathematical equipment (the portions of parallel system nodes involved for production rules extraction based on decision trees, associative rules and negative selection). The parameters of solved application task are also taken into account. They are the number of observations and the number of characteristics in a given set of data describing the results of observations of the object or process being studied. The Synthesized neural model is a polyalgorithmic. It allows estimating two characteristics of parallel computer system while executing the parallel method of production rules extraction. They are the time spent and the volume of memory used.

The practical value of the results obtained lies in the developed software which implements the proposed model and allows predicting the time spent and the volume of memory used of parallel computer system while solving practice tasks in which the parallel method of production rules extraction is realized.

ACKNOWLEDGMENTS

The work was performed as part of research work "Methods and means of computational intelligence and parallel computing for processing large amounts of data in diagnostic systems" (number of state registration 0116U007419) of software tools department of Zaporizhzhia National Technical University.

REFERENCES

1. Sobhani-Tehrani E. Fault diagnosis of nonlinear systems using a hybrid approach / E. Sobhani-Tehrani, K. Khorasani. – New York : Springer, 2009. – 265 p. – (Lecture notes in control and information sciences ; № 383). DOI: 10.1007/978-0-387-92907-1.

2. Price C. Computer based diagnostic systems / C. Price. – London : Springer, 1999. – 136 p. DOI: 10.1007/978-1-4471-0535-0.
3. Bodyanskiy Ye. Hybrid adaptive wavelet-neuro-fuzzy system for chaotic time series identification / Ye. Bodyanskiy, O. Vynokurova // Information Sciences. – 2013. – Vol. 220. – P. 170–179. DOI: 10.1016/j.ins.2012.07.044.
4. Ukil A. Intelligent Systems and Signal Processing in Power Engineering / A. Ukil. – Berlin : Springer, 2007. – 372 p.
5. Bishop C. M. Pattern recognition and machine learning / C. M. Bishop. – New York : Springer, 2006. – 738 p.
6. Бодянський Є. В. Послідовне нечітке кластерування на основі нейро-фаззі підходу / Є. В. Бодянський, А. О. Дейнеко, Я. В. Куценко // Радіоелектроніка, інформатика, управління. – 2016. – № 3. – С. 30–38. DOI: 10.15588/1607-3274-2016-3-4.
7. Analysis and design of intelligent systems using soft computing techniques / eds.: P. Melin, O. R. Castillo, E. G. Ramirez, J. Kacprzyk. – Heidelberg : Springer, 2007. – 855 p. DOI: 10.1007/978-3-540-72432-2.
8. Encyclopedia of machine learning / [eds. C. Sammut, G. I. Webb]. – New York : Springer, 2011. – 1031 p. DOI: 10.1007/978-0-387-30164-8.
9. Tenne Y. Computational Intelligence in Expensive Optimization Problems / Y. Tenne, C.-K. Goh. – Berlin : Springer, 2010. – 800 p. DOI: 10.1007/978-3-642-10701-6.
10. Duda R. O. Pattern Classification / R. O. Duda, P. E. Hart, D. G. Stork. – New Jersey : John Wiley & Sons, 2000. – 680 p.
11. Bow S. Pattern recognition and image preprocessing / S. Bow. – New York : Marcel Dekker Inc., 2002. – 698 c. DOI: 10.1201/9780203903896.
12. ASM handbook. – Vol. 17: Nondestructive evaluation and quality control. – Cleveland : ASM International, 1997. – 1607 p.
13. Oliinyk A. The decision tree construction based on a stochastic search for the neuro-fuzzy network synthesis / A. Oliinyk, S. A. Subbotin // Optical Memory and Neural Networks (Information Optics). – 2015. – Vol. 24, № 1. – P. 18–27. DOI: 10.3103/S1060992X15010038.
14. Oliinyk A. Association Rules Extraction for Pattern Recognition / A. Oliinyk, S. A. Subbotin // Pattern Recognition and Image Analysis. – 2016. – Vol. 26, № 2. – P. 419–426.
15. Diagnostic rule mining based on artificial immune systems for a case of uneven distribution of classes in sample / [S. Subbotin A. Oliinyk, V. Levashenko, E. Zaitseva] // Communications. – 2016. – Vol. 3. – P. 4–12.
16. Олійник А. О. Видобування продукційних правил на основі негативного відбору / А. О. Олійник // Радіоелектроніка, інформатика, управління. – 2016. – № 1. – С. 40–49. DOI: 10.15588/1607-3274-2016-1-5.
17. Gebali F. Algorithms and Parallel Computing / F. Gebali. – New Jersey : John Wiley & Sons, 2011. – 364 p. DOI: 10.1002/9780470932025.
18. Олійник А. О. Планування ресурсів паралельної обчислювальної системи при синтезі нейро-нечітких моделей для обробки великих даних / [А. О. Олійник, С. Ю. Скрупський, С. О. Суботін та ін.] // Радіоелектроніка, інформатика, управління. – 2016. – № 4. – С. 61–69. DOI: 10.15588/1607-3274-2016-4-8.
19. Sulistio A. Simulation of Parallel and Distributed Systems: A Taxonomy – and Survey of Tools / A. Sulistio, C. S. Yeo, R. Buyya // International Journal of Software Practice and Experience. Wiley Press. – 2002. – P. 1–19.
20. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml/.
21. Subbotin S. A. Synthesis of neuro-fuzzy models for the allocation and detection of objects on a complex background on the two-dimensional image / S. A. Subbotin // Computer modeling and intelligent systems : proceedings of the conference. – Zaporizhzhya : ZNTU, 2007. – P. 68–91.
22. Интеллектуальные информационные технологии проектирования автоматизированных систем диагностирования и распознавания образов : монография / [С. А. Субботин, Ан. А. Олейник, Е. А. Гофман и др.] ; под ред. С. А. Субботина. – Харьков : ООО «Компания Смит», 2012. – 317 с.
23. Oliinyk A. Training Sample Reduction Based on Association Rules for Neuro-Fuzzy Networks Synthesis / A. Oliinyk, T. Zaiko, S. Subbotin // Optical Memory and Neural Networks (Information Optics). – 2014. – Vol. 23, № 2. – P. 89–95. DOI: 10.3103/S1060992X14020039.
24. Bodyanskiy Ye. An evolving radial basis neural network with adaptive learning of its parameters and architecture / Ye. Bodyanskiy, A. K. Tyshchenko, A. Deineko // Automatic Control and Computer Sciences. – 2015. – Vol. 49, Issue 5. – P. 255–260. DOI: 10.3103/S0146411615050028.
25. Suzuki K. Artificial Neural Networks: Architectures and Applications / K. Suzuki. – New York : InTech, 2013. – 264 p. DOI: 10.5772/3409.
26. Quinn M. J. Parallel Programming in C with MPI and OpenMP / M. J. Quinn. – New York, NY : McGraw-Hill, 2004. – 529 p.
27. NVIDIA CUDA Compute Unified Device Architecture 5.5. Santa Clara: NVIDIA Corporation, 2014. – 117 p.

Article was submitted 24.02.2017.
After revision 20.03.2017.

Олійник А. О.[1], Скрупський С. Ю.[2], Шкарупило В. В.[2], Субботін С. О.[3]

[1]Канд. техн. наук, доцент кафедри програмних засобів, Запорізький національний технічний університет, Запоріжжя, Україна

[2]Канд. техн. наук, доцент кафедри комп'ютерних систем і мереж, Запорізький національний технічний університет, Запоріжжя, Україна

[3]Д-р техн. наук, професор, завідувач кафедри програмних засобів, Запорізький національний технічний університет, Запоріжжя, Україна

МОДЕЛЬ ОЦІНЮВАННЯ ВИКОРИСТОВУВАНИХ РЕСУРСІВ КОМП'ЮТЕРНОЇ СИСТЕМИ ПРИ ВИДОБУВАННІ ПРОДУКЦІЙНИХ ПРАВИЛ НА ОСНОВІ ПАРАЛЕЛЬНИХ ОБЧИСЛЕНЬ

Актуальність. Розглянуто задачу видобування продукційних правил при обробці великих масивів даних. Вирішено проблему оцінювання ресурсів комп'ютерної системи при видобуванні продукційних правил на основі паралельних обчислень. Об'єкт дослідження – процес видобування продукційних правил. Предмет дослідження – методи планування ресурсів паралельних комп'ютерних систем.

Мета роботи полягає в побудові моделі оцінювання ресурсів паралельних комп'ютерних систем при видобуванні продукційних правил.

Метод. Запропоновано модель оцінювання використовуваних ресурсів комп'ютерної системи при виконанні паралельного методу видобування продукційних правил. Синтезована модель враховує тип комп'ютерної системи, число процесів, на яких виконується задача, пропускну здатність мережі передачі даних, параметри використовуваного математичного забезпечення (частки вузлів паралельної системи, що використовуються для отримання продукційних правил на основі дерев рішень, асоціативних правил та негативного відбору, відповідно), а також параметри розв'язуваної прикладної задачі (кількість спостережень і кількість ознак в заданій множині даних, що описує результати спостережень за досліджуваним об'єктом або процесом). Синтезована нейромережева модель є поліалгорітмічною, дозволяє виконувати оцінювання двох характеристик (часу і обсягу споживаної пам'яті) паралельної системи при виконанні методу видобування продукційних правил.

Результати. Розроблено програмне забезпечення, яке реалізує запропоновану модель і дозволяє прогнозувати для паралельної системи витрачений час і обсяг споживаної пам'яті при вирішенні практичних завдань.

Висновки. Проведені експерименти підтвердили працездатність запропонованого математичного забезпечення і дозволяють рекомендувати його для використання на практиці при вирішенні задач обробки великих масивів даних. Перспективи подальших досліджень можуть полягати в створенні паралельних методів відбору інформативних ознак, а також експериментальному дослідженні запропонованої моделі на більшому комплексі практичних задач різної природи і розмірності.

Ключові слова: вибірка даних, паралельні обчислення, оцінювання ресурсів, продукційні правила, нейронна мережа.

Олейник А. А.[1], Скрупский С. Ю.[2], Шкарупило В. В.[2], Субботин С. А.[3]

[1]Канд. техн. наук, доцент кафедры программных средств, Запорожский национальный технический университет, Запорожье, Украина

[2]Канд. техн. наук, доцент кафедры компьютерных систем и сетей, Запорожский национальный технический университет, Запорожье, Украина

[3]Д-р техн. наук, профессор, заведующий кафедрой программных средств, Запорожский национальный технический университет, Запорожье, Украина

МОДЕЛЬ ОЦЕНИВАНИЯ ИСПОЛЬЗУЕМЫХ РЕСУРСОВ КОМПЬЮТЕРНОЙ СИСТЕМЫ ПРИ ИЗВЛЕЧЕНИИ ПРОДУКЦИОННЫХ ПРАВИЛ НА ОСНОВЕ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ

Актуальность. Рассмотрена задача извлечения продукционных правил при обработке больших массивов данных. Решена проблема оценивания ресурсов компьютерной системы при извлечении продукционных правил на основе параллельных вычислений. Объект исследования – процесс извлечения продукционных правил. Предмет исследования – методы планирования ресурсов параллельных компьютерных систем.

Цель работы заключается в построении модели оценивания ресурсов параллельных компьютерных систем при извлечении продукционных правил.

Метод. Предложена модель оценивания используемых ресурсов компьютерной системы при выполнении параллельного метода извлечения продукционных правил. Синтезированная модель учитывает тип компьютерной системы, число процессов, на которых выполняется задача, пропускную способность сети передачи данных, параметры используемого математического обеспечения (доли узлов параллельной системы, используемых для извлечения продукционных правил на основе деревьев решений, ассоциативных правил и отрицательного отбора, соответственно), а также параметры решаемой прикладной задачи (число наблюдений и число признаков в заданном множестве данных, описывающем результаты наблюдений за исследуемым объектом или процессом). Синтезированная нейросетевая модель является полиалгоритмичной, позволяет выполнять оценивание двух характеристик (времени и объема потребляемой памяти) параллельной системы при выполнении метода извлечения продукционных правил.

Результаты. Разработано программное обеспечение, которое реализует предложенную модель и позволяет прогнозировать для параллельной системы затраченное время и объем потребляемой памяти при решении практических задач.

Выводы. Проведенные эксперименты подтвердили работоспособность предложенного математического обеспечения и позволяют рекомендовать его для использования на практике при обработке больших массивов данных. Перспективы дальнейших исследований могут заключаться в создании параллельных методов отбора информативных признаков, а также экспериментальном исследовании предложенной модели на большем комплексе практических задач разной природы и размерности.

Ключевые слова: выборка данных, параллельные вычисления, оценивание ресурсов, продукционные правила, нейронная сеть.

REFERENCES

1. Sobhani-Tehrani E., Khorasani K. Fault diagnosis of nonlinear systems using a hybrid approach. New York, Springer, 2009, 265 p. (Lecture notes in control and information sciences, No. 383). DOI: 10.1007/978-0-387-92907-1.
2. Price C. Computer based diagnostic systems. London, Springer, 1999, 136 p. DOI: 10.1007/978-1-4471-0535-0.
3. Bodyanskiy Ye., Vynokurova O. Hybrid adaptive wavelet-neuro-fuzzy system for chaotic time series identification, *Information Sciences,* 2013, Vol. 220, pp. 170–179. DOI: 10.1016/j.ins.2012.07.044.
4. Ukil A. Intelligent Systems and Signal Processing in Power Engineering. Berlin, Springer, 2007, 372 p.
5. Bishop C. M. Pattern recognition and machine learning. New York, Springer, 2006, 738 p.
6. Bodyanskiy Ye. V., Deineko A. O., Kutsenko Ya. V. Sequential fuzzy clustering based on neuro-fuzzy approach*, Radio Electronics, Computer Science, Control*, 2016, No. 3, pp. 30–38. DOI: 10.15588/1607-3274-2016-3-4.
7. Eds. Melin P., Castillo O. R., Ramirez E. G., Kacprzyk J. Analysis and design of intelligent systems using soft computing techniques. Heidelberg, Springer, 2007, 855 p. DOI: 10.1007/978-3-540-72432-2.
8. Eds. Sammut C., Webb G. I. Encyclopedia of machine learning. New York, Springer, 2011, 1031 p. DOI: 10.1007/978-0-387-30164-8.
9. Tenne Y., Goh C.-K. Computational Intelligence in Expensive Optimization Problems. Berlin, Springer, 2010, 800 p. DOI: 10.1007/978-3-642-10701-6.
10. Duda R. O., Hart P. E., Stork D. G. Pattern Classification. New Jersey : John Wiley & Sons, 2000. – 680 p.
11. Bow S. Pattern recognition and image preprocessing. New York, Marcel Dekker Inc., 2002, 698 p. DOI: 10.1201/9780203903896.
12. ASM handbook. – Vol. 17: Nondestructive evaluation and quality control. Cleveland, ASM International, 1997, 1607 p.
13. Oliinyk A., Subbotin S. A. The decision tree construction based on a stochastic search for the neuro-fuzzy network synthesis, *Optical Memory and Neural Networks (Information Optics)*, 2015, Vol. 24, No. 1, pp. 18–27. DOI: 10.3103/S1060992X15010038.
14. Oliinyk A., Subbotin S. A. Association Rules Extraction for Pattern Recognition, *Pattern Recognition and Image Analysis*, 2016, Vol. 26, No. 2, pp. 419–426.
15. Subbotin S., Oliinyk A., Levashenko V., Zaitseva E. Diagnostic rule mining based on artificial immune systems for a case of uneven distribution of classes in sample, *Communications,* 2016, Vol. 3, pp. 4–12.
16. Oliinyk A. Production rules extraction based on negative selection, *Radio Electronics, Computer Science, Control,* 2016, No. 1, pp. 40–49. DOI: 10.15588/1607-3274-2016-1-5.
17. Gebali F. Algorithms and Parallel Computing. New Jersey, John Wiley & Sons, 2011, 364 p. DOI: 10.1002/9780470932025.
18. Oliinyk A., Skrupsky S., Subbotin S., Blagodariov O., Gofman Ye. Parallel computing system resources planning for neuro-fuzzy models synthesis and big data processing, Radio Electronics, Computer Science, Control, 2016, Vol. 4, pp. 61–69. DOI: 10.15588/1607-3274-2016-4-8.
19. Sulistio A., Yeo C. S., Buyya R. Simulation of Parallel and Distributed Systems: A Taxonomy – and Survey of Tools, *International Journal of Software Practice and Experience*. Wiley Press, 2002, pp. 1–19.
20. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml/.
21. Subbotin S.A. Synthesis of neuro-fuzzy models for the allocation and detection of objects on a complex background on the two-dimensional image, *Computer modeling and intelligent systems : proceedings of the conference*, Zaporizhzhya, ZNTU, 2007, pp. 68–91.
22. Subbotin S. A., Oliinyk A., Gofman Ye., Zaitsev S., Oliinyk O. Intelligent information technology of automated diagnostic and pattern recognition systems development : monograph. Kharkov, “Company Smith”, 2012, 317 p. (In russian).
23. Oliinyk A., Zaiko T., Subbotin S. Training Sample Reduction Based on Association Rules for Neuro-Fuzzy Networks Synthesis, *Optical Memory and Neural Networks (Information Optics),* 2014, Vol. 23, No. 2. pp. 89–95. DOI: 10.3103/S1060992X14020039.
24. Bodyanskiy Ye., Tyshchenko A. K., Deineko A. An evolving radial basis neural network with adaptive learning of its parameters and architecture, *Automatic Control and Computer Sciences,* 2015, Vol. 49, Issue 5, pp. 255–260. DOI: 10.3103/S0146411615050028.
25. Suzuki K. Artificial Neural Networks: Architectures and Applications. New York, InTech, 2013, 264 p. DOI: 10.5772/3409.
26. Quinn M. J. Parallel Programming in C with MPI and OpenMP. New York, NY: McGraw-Hill, 2004, 529 p.
27. NVIDIA CUDA Compute Unified Device Architecture 5.5. Santa Clara: NVIDIA Corporation, 2014, 117 p.