UDC 004.272.26: 004.93

Oliinyk A. O.[1], Skrupsky S. Yu.[2], Shkarupylo V. V.[3], Blagodariov O. Yu.[4]

[1]PhD., Associate Professor of Department of Software Tools, Zaporizhzhia National Technical University, Zaporizhzhia, Ukraine
[2]PhD, Associate Professor of Computer Systems and Networks Department, Zaporizhzhia National Technical University, Zaporizhzhia, Ukraine
[3]PhD, Associate Professor of Computer Systems and Networks Department, Zaporizhzhia National Technical University, Zaporizhzhia, Ukraine
[4]Postgraduate Student of Department of Software Tools, Zaporizhzhia National Technical University, Zaporizhzhia, Ukraine

PARALLEL MULTIAGENT METHOD OF BIG DATA REDUCTION FOR PATTERN RECOGNITION

Context. The problem of feature selection for big data processing based on the multi-agent approach and parallel computation has been solved. The object of research is the process of feature selection. The subject of the research are the methods of feature selection.

Objective. The purpose of the work is to create a parallel multi-agent method for reducing of big data sets.

Method. The article deals with the parallel multi-agent method for reducing of big data sets. The developed method involves splitting multiple agents into several subsets for parallel search of an informative combination of features in different areas of the search space. At the same time, it is suggested that the parallel nodes of the computer system perform the most resource-intensive operations associated with estimating the current set of agents, as well as the need to create and modify new sets of solutions based on stochastic computations. This allows to speed up the process of multi-agent search of informative combination of features, as well as to reduce the practical threshold for application of the multi-agent method with indirect communication between agents for reducing big data sets.

Results. The software which implements the proposed method and allows to select informative features based on the multi-agent approach and parallel computation has been developed.

Conclusions. The conducted experiments have confirmed the proposed software operability and allow recommending it for use in practice for solving the problems of big data processing for pattern recognition. The prospects for further research may include the modification of the developed parallel method for feature selection by using different criteria for estimation of the group information of features, as well as an experimental study of proposed method on more complex practical problems of different nature and dimensionality.

Keywords: agent, data set, feature selection, parallel computing, multi-agent approach, pattern recognition.

NOMENCLATURE

A – set of agents;

CPU – Central Processing Unit;

GPU – Graphical Processing Unit;

G_k – value of objective function of k-th agent;

$J(Xe)$ – criterion for assessing the significance of a set of features Xe;

M – number of features in the sample of observations S;

N_{er} – number of incorrectly recognized (classified) observations of the sample $S =< P, T >$ on the synthesized model;

N_{pr} – number of processes, on which task is performed;

N_{χ} – total number of agents in set A;

p_{qm} – value of m-th feature (attribute) for q-th observation ($m = 1, 2, ..., M$, $q = 1, 2, ..., Q$);

Q – number of observations in the given sample of observations S;

S – sample of observations (training sample);

t_q – value of output parameter of q-th observation;

$t_{q\,\mathrm{mod}}$ – value of output parameter of q-th observation, calculated from the synthesized model;

T – set of output parameter values;

χ_k – k-th agent in set A;

XS – set of all possible combinations of features, obtained from the initial set of characteristics P.

INTRODUCTION

The development of automated systems for pattern recognition is associated with the need of big data processing [1–6]. Typically, the original samples of data describing the objects or processes under investigation may contain redundant and uninformative information [7–13]. The use of such information in the synthesis of recognition models leads to an increase of their complexity and redundancy, as well as a reduction in their generalizing abilities. Therefore, before synthesis of recognition models, it is relevant to perform preprocessing of data in order to exclude uninformative and redundant features from training samples [3, 5, 8].

Typically, well-known methods of feature selection use a greedy search strategy [6, 8], which often doesn‘t allow choosing the most informative combination of features. Stochastic methods [6, 12] are highly iterative and require substantial outlays of computing and time resources, making them difficult to use in practice. A multiagent method for feature selection that doesn‘t use greedy search strategy, based on modeling of agents movement in search space through stochastic computations, is offered in [14]. However, this method no longer adequate to handle big data sets due to the high iterative and consistent nature of the calculations.

It is therefore appropriate to parallelize the most computationally complex and resource-consuming operations of the multi-agent method of feature selection [14], which will reduce the practical threshold for applicability of such method when big data processing.

The purpose of the work is to create a parallel multi-agent method for big data sets reducing.

1 PROBLEM STATEMENT

Let there is a set of observations S (1):

$$S =< P, T > . \quad (1)$$

DOI 10.15588/1607-3274-2017-2-9

Then the problem of informative features selection in an idealized formulation [8, 15–17] can be represented as: find a combination of features X^* from the original data set $S =< P, T >$, at which the minimum of given criterion (2) for assessing the quality of feature set is achieved:

$$J(X^*) = \min_{Xe \in XS} J(Xe). \quad (2)$$

The error of the synthesized model, normally, is used as an criteria for assessing the significance of features set $J(Xe)$ [4–6]:

– recognition error (in problems with discrete output T) [4–6], calculated according to the formula (3):

$$E = \frac{N_{er}}{Q}; \quad (3)$$

– standard error (in the case where the output parameter T can take real values from a certain range $T \in [t_{\min}; t_{\max}]$) [6], calculated according to the formula (4):

$$E = \frac{1}{Q} \sum_{q=1}^{Q} \left(t_q - t_{q\,\mathrm{mod}}\right)^2. \quad (4)$$

In calculating criteria values (3) and (4), the model is synthesized based on the data of the training sample S, using only the features that correspond to the combination Xe.

2 LITERATURE REVIEW

Currently, there are various methods of reducing the dimension of the feature space [3–6, 15–17]: brute-force method, depth-first search, breadth-first search, branch and bound, group method of data handling, method of sequential addition of features, method of sequential removal of features, method of alternately adding and removing of features, ranking of features, features clustering, method of random search with adaptation and evolutionary search for the selection of features [15–17].

Usually, in solving the problems of recognition, there is a system of statistically dependent features, whose set informativity isn't expressed through the informativeness of individual features. Therefore, in solving practical problems, it is necessary to evaluate a set of features, rather than each attribute separately. Thus, the use of ranking criteria on the computed individual assessment is unacceptable [6].

The application of brute-force method requires the evaluation of all possible combinations of features made up of the original data, which makes it impossible to use this approach with a large number of features in the source set, since it requires huge computational costs.

Heuristic search methods [15, 17] are not effective enough, because of sub-optimal of a greedy search strategy involving the sequential addition or deletion of one feature at each iteration, therefore of which the resulting set of characteristics contains redundant features that correlate with other features in the set.

It seems expedient to use the methods of stochastic search (in particular, the multi-agent approach) [13, 14, 18], to search such combination of informative features under conditions of mutual dependence on each other. Since such methods are more suitable for finding new solutions by combining the best solutions that were obtained at different iterations and have the opportunity to exit from local optima.

The proposed multi-agent method [14] with an indirect communication between agents allows to select the most significant features. However, the method takes considerable time in processing of big data, because it is highly iterative and provides for sequential implementation of calculations. To address these drawbacks it is advisable to parallelize the multi-agent search of most meaningful combination of features in the processing of big data sets.

3 MATERIALS AND METHODS

In the developed parallel multi-agent method for big data reducing, it is proposed to split a set of agents $A = \{\chi_1, \chi_2, ..., \chi_{N_\chi}\}$ into several subsets for parallel search of informative combination of features in different areas of search space. Meanwhile, to speed up the process of multi-agent search for informative combination of features in the developed method, it is suggested to fulfil the most resource-intensive operations related with evaluation of the current set of agents, including the need to create and modify of new sets of solutions based on stochastic computations on the nodes of the parallel computing system.

As noted previously, in the proposed parallel multi-agent method for big data reducing after initialization phase, a set of agents $A = \{\chi_1, \chi_2, ..., \chi_{N_\chi}\}$ is split into several subsets $A^{(1)}, A^{(2)}, ..., A^{(Npr)}$ (5)

$$A \to \left\{A^{(1)}, A^{(2)}, ..., A^{(Npr)}\right\}. \quad (5)$$

The total number of agents N_χ in the partition $A \to \left\{A^{(1)}, A^{(2)}, ..., A^{(Npr)}\right\}$ remains unchanged (6):

$$\left|A^{(1)}\right| + \left|A^{(2)}\right| + ... + \left|A^{(Npr)}\right| = N_\chi. \quad (6)$$

The partitioning $A \to \left\{A^{(1)}, A^{(2)}, ..., A^{(Npr)}\right\}$ is done so that to ensure the separation of groups of agents $A^{(j)}$, over the search space with view to more detailed investigation of its various areas. For selected groups of compactly located agents $A^{(j)}$, it is proposed to apply cluster analysis methods [6, 9, 16].

After that, in each of the received subsets of agents $A^{(j)}$ ($j = 1, 2, ..., N_{pr}$), it is proposed to conduct a multi-agent search to select informative features from the given data samples $S =< P, T >$, periodically checking the stopping criteria and, if necessary, combining the subsets $A^{(j)}$.

In order to increase the efficiency of the multi-agent search for combination of informative features (reducing the search time), it is expedient to parallelize the most resource-intensive operations.

As noted above, the multi-agent method of feature selection with indirect link between agents, involves the implementation of initialization, chemotaxis modeling, reproduction, elimination and dispersion, checking the stopping criteria, and restarting agents.

In the initialization phase, the main parameters of method are defined, and the begin coordinates of agents χ_k ($k = 1,2,\dots,N_\chi$) are randomly generated, after which values of the objective function $G_k = G(\chi_k)$ are calculated. This stage doesn't involve complex iterative computational procedures, so it is proposed to perform it on the main stream. Note that with a large number of agents N_χ, as well as processing complex samples $S =< P,T >$, you can parallelize the process of assessing the initial positions of agents χ_k (calculating the values of the objective function $G_k = G(\chi_k)$, $k = 1,2,\dots,N_\chi$).

The stage of chemotaxis modeling is connected with iterative implementation of tumbling, moving and sliding operators for each agent, assuming the calculation of new values of the coordinates of agents $\chi_k^{(t+1)}$ in the search space. In addition, at this stage, the evaluation of new agent's positions $G_k^{(t+1)} = G\left(\chi_k^{(t+1)}\right)$, suggesting the need to data sample $S =< P,T >$ for each of the agents. The complexity of the computational procedures of this stage necessitates its parallel implementation for the possibility of performing multi-agent search for informative combination of features in subset of agents $A^{(j)}$ at the *j*-th node of the parallel system.

At the stage of reproduction, new set of agents $A = \{\chi_1, \chi_2, \dots, \chi_{N\chi}\}$ is created by performing stochastic calculations. At the same time, complex procedures are performed to select the most suitable agents, select the parent agent pair, and directly generate new agents (new coordinate points in the search space).

The stage of exclusion and dispersion also involves the need to process the entire current set of agents $A = \{\chi_1, \chi_2, \dots, \chi_{N\chi}\}$ in order to randomly change the coordinates of some agents to exit from local extremums.

Therefore, for the possibility of implement multi-agent optimization, it is suggested that the nodes of parallel computing system perform the stages of reproduction, elimination and dispersion, as well as checking the termination criteria. These stages are associated with the need to create and modify new sets of solutions based on stochastic computations and, together with the chemotaxis simulation stage, make it possible to search for a combination of informative features in each of the subsets N_{pr} of agents $A^{(j)}$ ($j = 1,2,\dots, N_{pr}$).

Based on the mentioned above, we present a model of the multiagent search process for a combination of informative features in the form of Fig. 1.

One iteration of the multi-agent search of an informative combination of features $MAS(A^{(j)})$ among a set of agents $A^{(j)}$ on the *j*-th node of a computer system ($j = 1,2,\dots, N_{pr}$) is schematically shown in Fig. 2. In so doing, the following notation is used in the Fig. 2: $Chem\left(A^{(j)}\right)$ is a chemotaxis modeling operator over multiple agents $A^{(j)}$; $Cross\left(A^{(j)}\right)$ is a reproduction operator; $Disp\left(A^{(j)}\right)$ is an exclusion and scattering operator; $Crit\left(A^{(j)}\right)$ is an operator of checking the criteria for the completion of multi-agent search $MAS(A^{(j)})$.

As can be seen from Fig. 1, computationally complex operations that are decided to parallelize, are related with operation of chemotaxis simulation, reproduction, exclusion and dispersion. The choice of these operations for parallel implementation of calculations is also due to the fact that information which is easily amenable to parallelization and associated with the current set of agents (coordinates of points χ_k in the search space and the corresponding values of the objective function $G_k = G(\chi_k)$) is iteratively processed on them. In addition, on the nodes of the parallel computer system are also offered to perform a check of the stopping criteria in order to cessation of the multi-agent optimization in case an optimal result is obtained (a set of characteristics that is acceptable for the solution of the current practical task). Sequences of initialization and restart of agents are consistently performed.

It is suggested that after performing multi-agent search at N_{it} iterations on nodes of parallel computer system, the information about the current sets of agents $A^{(j)}$ ($j = 1,2,\dots, N_{pr}$) and the values of their cost functions $G^{(j)}$ on the main process be transmitted. As a result of which, sets A (7) and G (8) are formed:

$$A = \bigcup_{j=1}^{N_{pr}} A^{(j)} = \{\chi_1, \chi_2, \dots, \chi_{N\chi}\}, \tag{7}$$

$$G = \bigcup_{j=1}^{N_{pr}} G^{(j)} = \{G_1, G_2, \dots, G_{N\chi}\}. \tag{8}$$

Then, the process of restarting agents is performed on the main process. In this case, unlike the known multi-agent method with indirect connection between agents BFO [18], a new set of agents is proposed to be formed in accordance with expressions (9) and (10):

$$A = A_{elite} \cup A_{rnd} \cup A_{cross}, \tag{9}$$

$$|A_{elite}| + |A_{rnd}| + |A_{cross}| = N_\chi, \tag{10}$$

where A_{elite} is a set of elite solutions χ_k (best agents by the value of the objective function G_k) in each of the subsets $A^{(j)}$: $A_{elite} = \{\chi_{elite1}, \chi_{elite2}, \dots, \chi_{eliteN_{pr}}\}$,

(

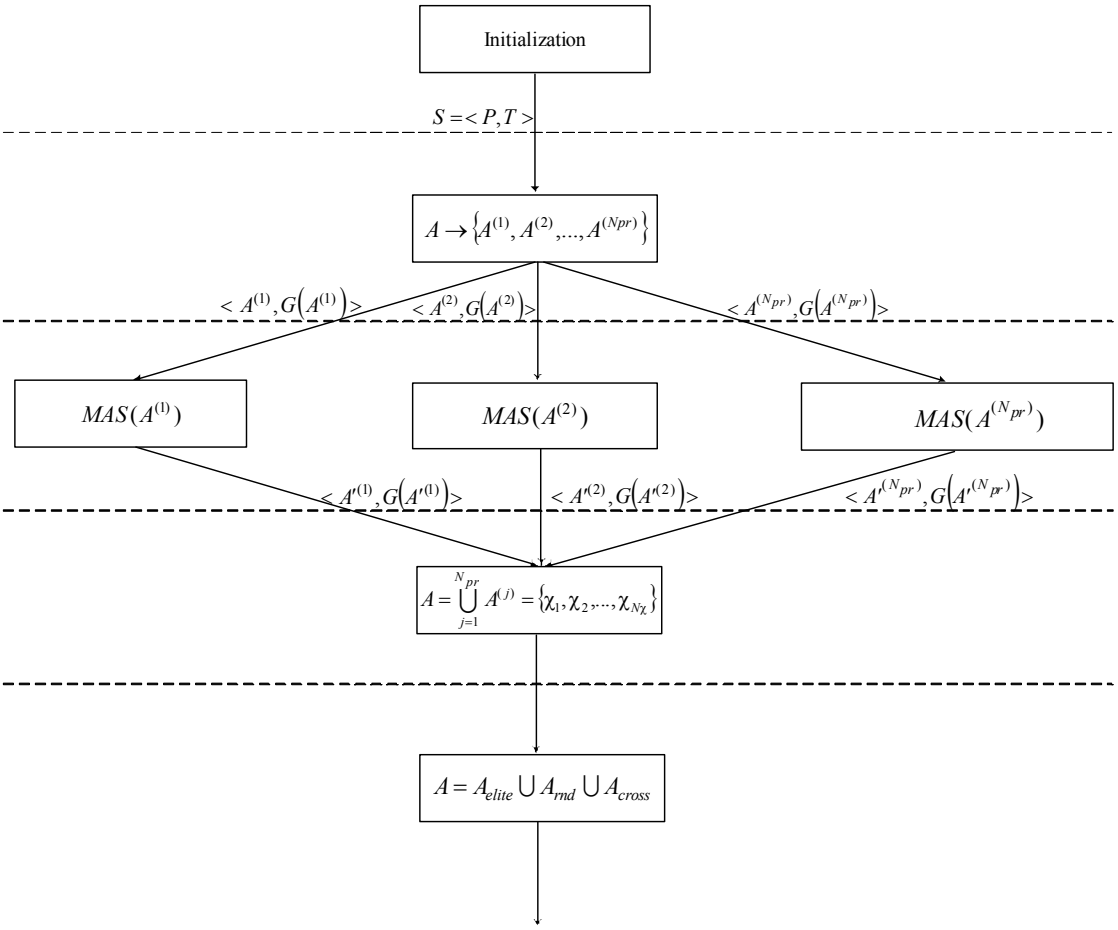

Figure 1 – The model of the multiagent search process for a feature selection

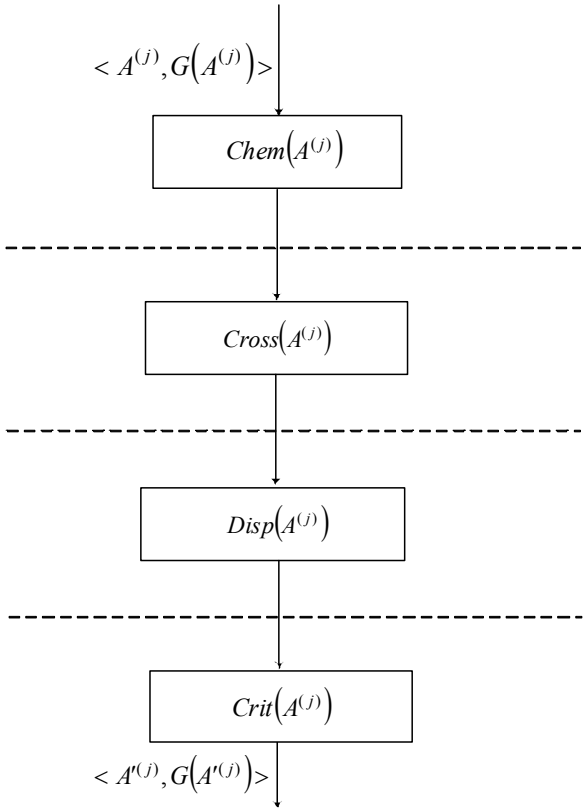

Figure 2 – One iteration of multi-agent search $MAS(A^{(j)})$ in a set of agents $A^{(j)}$

$A_{rnd} = \{\chi_{rnd1}, \chi_{rnd2}, ..., \chi_{rnd|A_{rnd}|}\}$, $\chi_{rnd\,j} = \underset{\chi_k \in A^{(j)}}{\text{rand}}(\chi_k)$, $|A_{rnd}| = \alpha_{rnd} N_\chi$, α_{rnd} is a fraction of agents in the new set $A = \{\chi_1, \chi_2, ..., \chi_{N\chi}\}$, randomly selected from the previous set of solutions. If $|A_{rnd}| > N_{pr}$ then random selection occurs successively among each of N_{pr} the subsets $A^{(j)}$, after which the process continues until the required number of solutions $|A_{rnd}|$ is reached, while the subset $A^{(j)}$ is randomly selected without repeating;

A_{cross} is a set of solutions χ_k, generated using a reproduction operator based on the current set of agents $A = \{\chi_1, \chi_2, ..., \chi_{N\chi}\}$. It is proposed to use a reproduction operator similar to that proposed in [14], using the evolutionary operators of proportional selection, arithmetic crossing and simple mutation [6, 8, 14];

$|A_{elite}|, |A_{rnd}|, |A_{cross}|$ are the total numbers of agents in sets A_{elite}, A_{rnd} and A_{cross}, accordingly.

Therefore, new set of agents $A = \{\chi_1, \chi_2, ..., \chi_{N\chi}\}$ in the proposed method is formed from elite agents A_{elite}, randomly selected agents from the previous set of decisions A_{rnd}, and also agents obtained by applying the reproduction operator A_{cross}. Such approach allows not only to save the current best solutions found in the search process, but also to provide a way out of local optima and the possibility of exploring different areas of search space.

The proposed parallel multiagent method of big data reduction involves splitting multiple agents into several subsets for parallel search of informative combination of features in different areas of the search space. At the nodes of parallel computing system, it is suggested to perform the most resource-intensive operations related to the evaluation of the current set of agents, as well as the need to create and modify new sets of solutions based on stochastic computations. This makes it possible to speed up the process of multi-agent search for informative combination of features, as well as to reduce the practical threshold for using a multi-agent method with indirect communication between agents to reduce big data sets.

Moreover, the proposed method fits well to SIMD (single instruction multiple data) architecture, because after the agent initialization stage the branches of algorithm perform the same actions on multiple data taking into account the stochastic component, which is data too. Hence, GPU implementation of the proposed method will probably demonstrate good computational process speedup.

4 EXPERIMENTS

The developed parallel multi-agent method of big data sets reduction has been implemented in C language using the MPI and CUDA libraries. Besides, the data exchange between the main core, which performed the initialization and agent restart stages, and other cores of cluster nodes has been performed by multiple exchange functions of MPI library (Bcast, Gather, Scatter, Reduce). The GPU cores used the global memory of GPU, in which many agents were placed. Each core of GPU performed a separate branch of the algorithm, and since the number of cores in the GPU is counted as many hundred, as a result, a significant number of evaluations of feature combinations were performed in one cycle.

To check the effectiveness of the proposed method with the help of the developed software, a problem of selecting informative features for the recognition of vehicles was solved [19, 20]. The training sample contained information on 10,000 images obtained from highways, taken in gray. Using the recognition system [20], images of interest areas with a vehicle were identified on the images, which were manually classified by an expert person (recognition was carried out in the following classes: 0 – not recognized, 1 – motorcyclist, 2 – car, 3 – truck, 4 – bus, 5 – minivan or minibus) and were displayed into a matrix 128*128 (16384 points). The resulting images were transformed by calculating 26 characteristics that generalize graphic information about object [20]. For each class of vehicle, a different recognition model has been synthesized, which makes it possible to determine whether the recognizable means belong to this type. Thus, five training samples $S = < P, T >$ of 10,000 instances were obtained, each of which was characterized by 26 features. The task was to reduce the number of attributes of training samples to identify the most informative combinations consisting of no more than 12 features. In the tables below, the results of selection of characteristics for the synthesis of recognition models defining the belonging of a motor vehicle to class 2 “passenger car” are presented.

To provide the experiments the following equipment has been involved:

– the cluster of Pukhov Institute for Modeling in Energy Engineering NAS of Ukraine, which involved 16 logical nodes (cores), each of which performed one process. The cluster configuration is as follows: processors Intel Xeon 5405, RAM – 4Ч2 GB DDR-2 for each node, communication environment InfiniBand 20Gb/s, middleware Torque and OMPI;

– NVIDIA GTX 960 with 1024 thread’s processors.

To compare the proposed method with existing analogues, the separation of combination of features was performed using various methods: heuristic search, the method of main components analyzing , the method of group accounting of arguments, the canonical model of genetic search, multi-agent methods for selecting informative features with direct and indirect communication between agents and using the developed parallel multi-agent method of informative features selection with indirect communication between agents. Values of parameters of the multi-agent method with indirect communication between the agents were: $N_\chi = 100$; $N_{re} = 4$; $N_s = 4$; $N_c = 20$; $N_{ed} = 2$; $P_{ed} = 0{,}25$; $d_{attract} = 0{,}1$; $w_{attract} = 0{,}2$; $h_{repellant} = d_{attract}$; $w_{repellant} = 10$ [14, 18]. Since the optimization process of most of the listed methods is of a stochastic nature, the search for optimal solutions

during experiments was carried out 100 times, after which the average values of investigated parameters on basis of the obtained results were calculated. As criteria for assessing effectiveness of the selection of features followed were used [14, 18]:

– the number of accesses to the objective function N_{fit} necessary to achieve the result with required accuracy;

– the error of the method E, calculated by the formula (3);

– the operating time of the method T, necessary to achieve an acceptable solution;

– the number of features of k, selected by the method as informative.

As the objective function E, the probability of making erroneous decisions (3) for a neural network of direct propagation, containing 3 neurons on the first layer and one neuron on the second layer was used. The neural network was synthesized on the basis of the appropriate combination of characteristics of the sample $S = < P, T >$, the neuroelements had a logistic sigmoid activation function. As discriminant functions weighted sums were used.

The proposed method was performed on 1, 2, 4, 8, 12, 16 cluster nodes and the time spent on the method was recorded. In addition, speedup of the computational process and efficiency of the computer system were calculated. Moreover, communication overheads (transfers and synchronizations) were analyzed.

Similar experiments were performed using the GPU.

5 RESULTS

Table 1 presents the characteristic values of the proposed and known methods of informative features selection in the recognition of vehicles.

Figures 3 and 4 shows dependences between the proposed method execution time in seconds (*Tspent*) and the number of involved cluster nodes (fig. 3), and the number of GPU threads (fig. 4).

Table 1 – Characteristics of selection features methods for vehicles recognition

№	Method of selection of features	Values of comparison criteria			
		E	N_{fit}	T	k
1	Method of analyzing the main components (MAMC) [6, 15]	0.0412	–	524	12
2	Method of group accounting of arguments (MGAA) [5, 8, 17]	0.0358	9761	18578	11
3	Canonical method of evolutionary search (CMES) [6, 19]	0.0219	10000	23171	10
4	Method of alternately adding and removing of features (MARF) [6, 8]	0.0471	871	2199	12
5	Multiagent method with indirect connection between agents (MMICA) (Ant Colony Optimization for Feature Selection) [19]	0.0198	4872	10318	10
6	Multiagent method with direct connection between agents (MMDCA) (Bee Colony Optimization for Feature Selection) [19]	0.0191	4719	10192	11
7	Parallel multiagent method of big data sets reduction (PMMBDSR), single core implementation $N_{pr} = 1$	0.0172	5418	11461	10
	PMMBDSR, implementation in $N_{pr} = 12$		65017	1868	
	PMMBDSR, implementation in 60 threads of GPU		48216	7317	

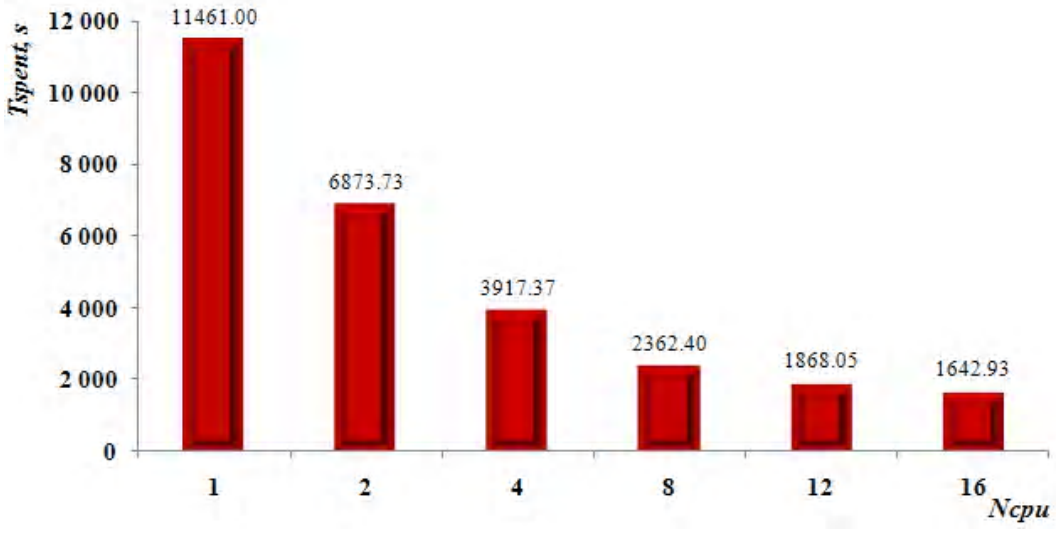

Figure 3 – Dependence between the proposed method execution time and the number of involved cluster nodes

In Figures 5 and 6 the speedup graphs of the computational process $Speedup_{pr}$ on the cluster and on the GPU, respectively, were shown.

Graph of the cluster efficiency $Efficiency_{pr}$ when performing the proposed method is shown in Figure 7.

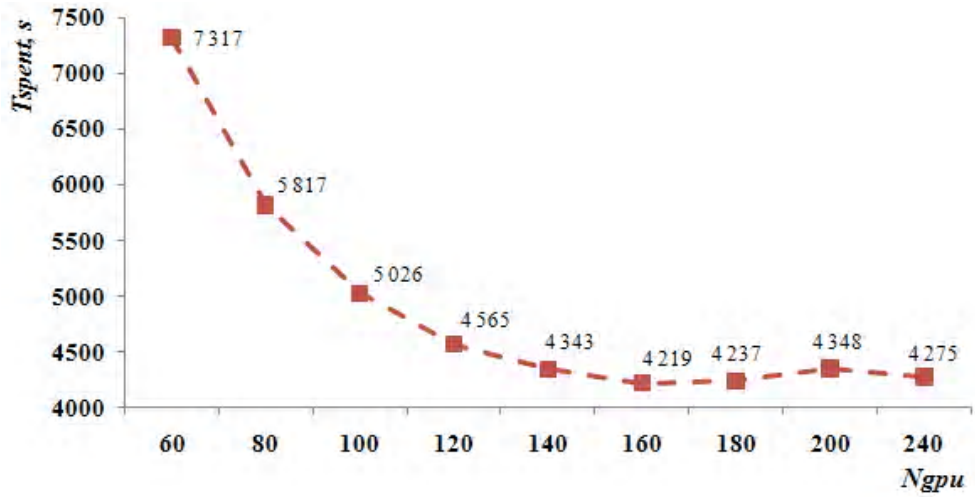

Figure 4 – Dependence between the proposed method execution time and the number of involved GPU threads

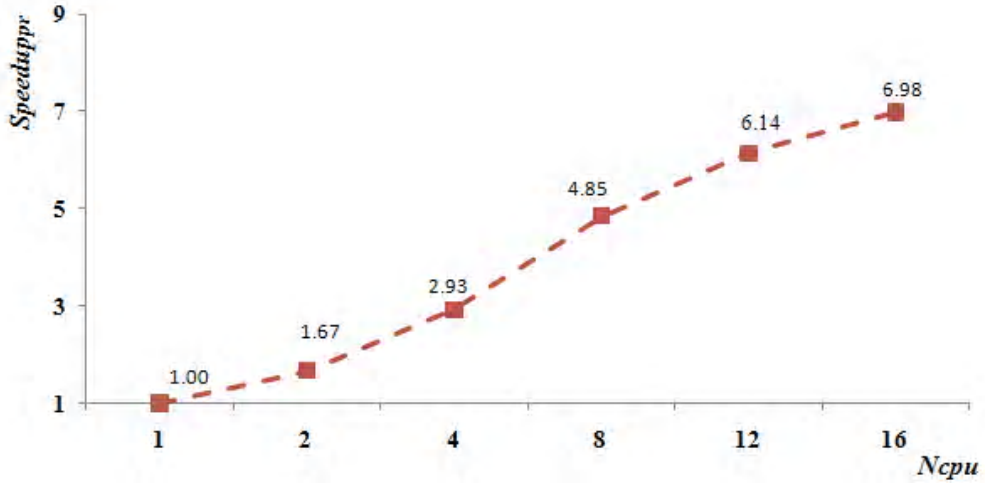

Figure 5 – Dependence between the computational speedup and the number of involved cluster nodes

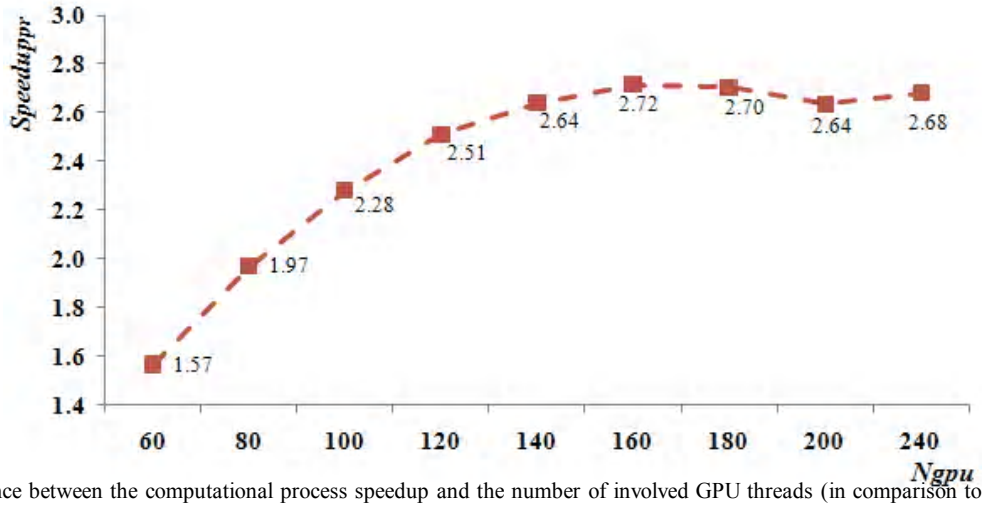

Figure 6 – Dependence between the computational process speedup and the number of involved GPU threads (in comparison to one cluster node)

As a result of the analysis of reasons of computer system efficiency decrease the corresponding graphs were formed. They show percentage of communication overheads *Overhead* (transfers and synchronizations) in the computational process from the number of involved cluster nodes (Figure 8) and GPU threads (Figure 9).

Graph of execution time *Tspent* of the proposed method from the number of agents, processed on each cluster core (Agents Per cluster Core) *ApC* is shown in the Figure. 10. The *ApC* criterion was calculated by the formula (9):

$$ApC = \frac{N_{\chi}}{N_{pr}}. \quad (9)$$

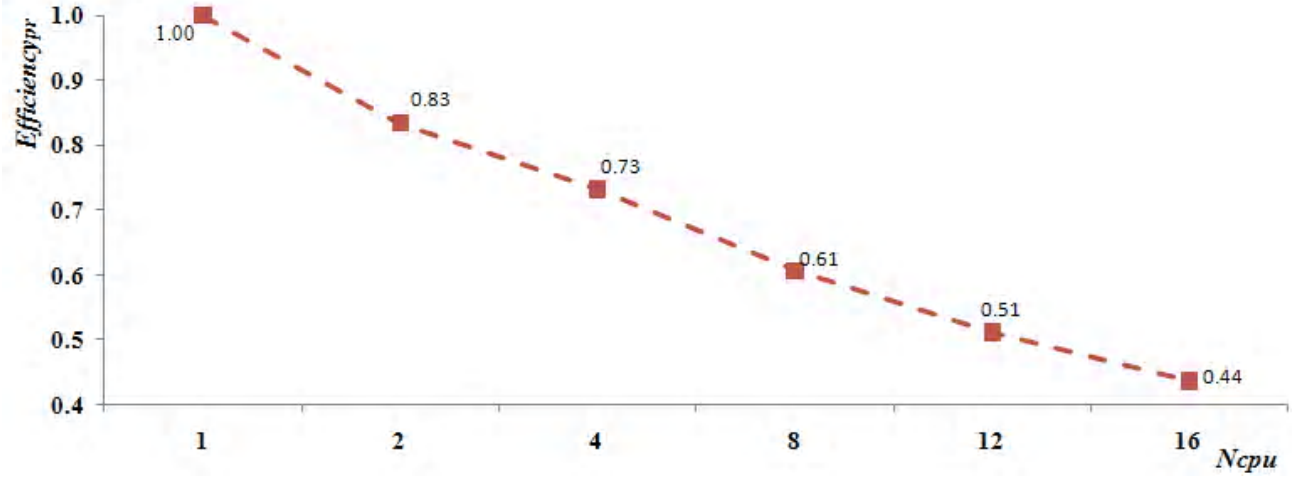

Figure 7 – Graph of the cluster efficiency when performing the proposed method

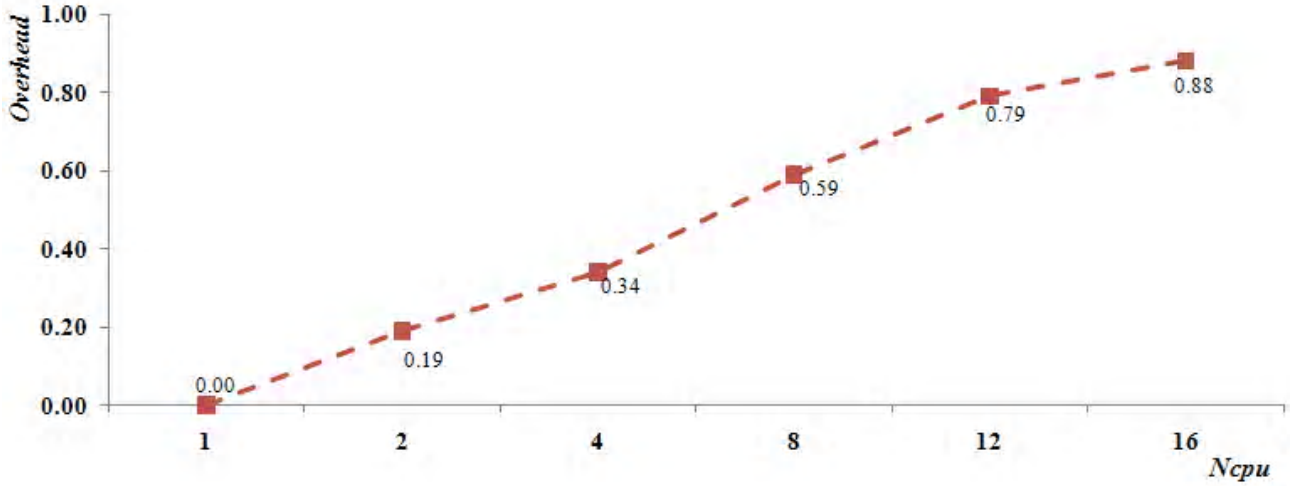

Figure 8 – Graph of communication overhead from the number of cluster nodes involved

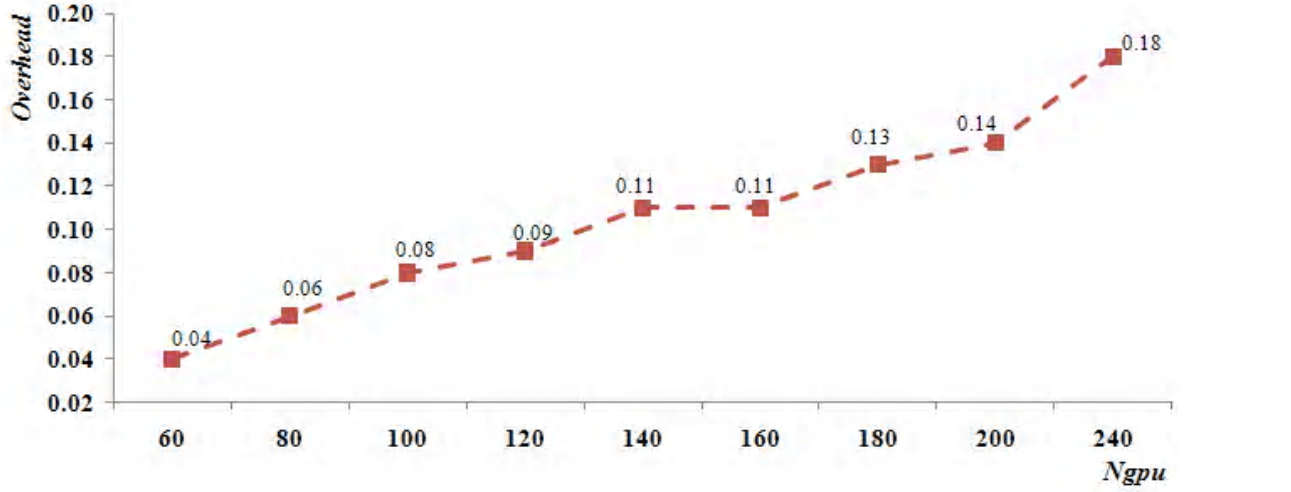

Figure 9 – Graph of communication overhead from the number of GPU threads involved

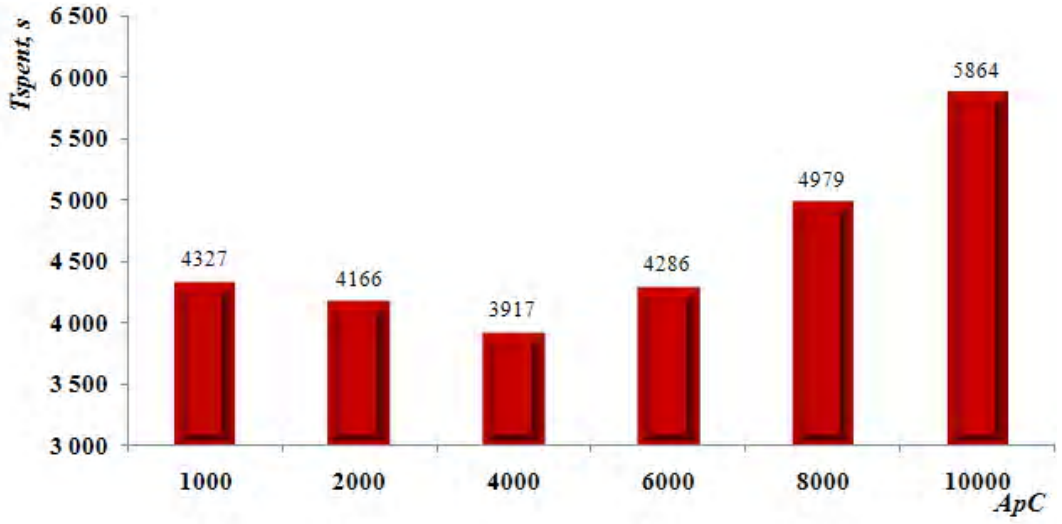

Figure 10 – Execution time graph of the proposed method from the number of agents per cluster core

6 DISCUSSION

As shown in Table 1 combinations of features which obtained using stochastic methods (CMES, MMICA, MMDCA, PMMBDSR) were characterized by more acceptable values of the objective function (for example, $E = 0.0172$ for PMMBDSR method) compared with the MAMC, MGAA and MARF methods (error value is varied in the range of $E = 0.0358$ for MGAA to $E = 0.0471$ for MARF). Such results are explained by a more effective analysis of search space using stochastic methods, including developed PMMBDSR method.

The operating time T of the methods varies from 524 s (for MAMC) to 23171 s (for CMES), which is related to number of calculations of the values of the objective function before the search stops. Such large operating time values of the selecting features methods are associated with the need to construct a recognition model based on each of the evaluated combinations of features. Obviously, sequential implementations of stochastic methods investigate a larger number of points in the search space, because of the stochastic nature, which causes large time consumption for their operation.

The results of Table 1 confirmed advisability of parallelizing the proposed method PMMBDSR: with number of parallel processes $N_{pr} = 12$, operation time of the method is 1868 s, that is less than the time of the heuristic method MARF ($T = 2199$ s)

The use of stochastic (evolutionary and multi-agent) methods allowed to select combinations that consist of fewer features ($k = 10$ for CMES, MMICA and PMMBDSR) compared with the use of other methods ($k = 11$ and 12 for other methods). This also indicates a more effective investigation of the features space using stochastic methods.

Consequently, the obtained values of the effectiveness estimation criteria of methods for selection informative features (E, N_{fit}, T, k) indicate the advisability of applying the proposed method for solving practical recognition problems.

Analysis of Figures 3–6 shows that the proposed method is well parallelized on a cluster and on a graphics processor. For instance, on 16 nodes of the cluster speedup of the computational process was 6.98, that allowed reducing the execution time of the method in computer system from 11461 s to 1643 s. On GPU, speedup of computational process compared to one core of the cluster was from 1.57 to 2.68, depending on the number of involved GPU threads.

Figure 7 shows a significant efficiency decrease of the computer system that performs the proposed method with an increase of the number of involved nodes. This is associated with an increase of overhead portion (synchronizations and transfers) with each new involved node of the system. Figure 8 confirms this fact. For example, when using 4 cluster nodes, the overhead portion in the general computing process is 0.34 and when using 16 cluster nodes is 0.88. As a result, growth in the number of involved nodes by 4 times (from 4 to 16) increases the speedup not linearly by 4 times, but only by about 2.38 times (Figure 5). On the GPU, the overheads increase with the number of involved threads not as significant as on the cluster (Figure 9). However, the capability of GPU in implementing of the proposed method is limited by the frequencies of the stream processors and by the bandwidth of data buses. As a result, the GPU managed to achieve the execution time of the proposed method, comparable to the four nodes of cluster (Figures 3, 4), which is an acceptable result.

The analysis of the runtime graph of the proposed method from the number of agents to the cluster core (Figure 10) shows that when the system's load is less than 4 000 agents per cluster core, the system is not fully loaded. For 4000 agents per core, the computer system is used efficiently, it finds a solution earlier than with a lesser load.

With a rise in the number of agents per core of more than 4 000, the time for finding solutions rises due to an increase in the overhead percentage. A concrete number of agents (in this case 4000), by which the system is used effectively depends on the capability of specific equipment. However, the form of the graph (as in Figure 10) should be preserved on other clusters.

CONCLUSIONS

The actual task of automation of large data sets reduction process based on the multi-agent approach has been solved

Scientific novelty lies in the fact that the parallel multiagent method of big data sets reduction has been proposed. The developed method involves splitting multiple agents into several subsets for parallel search of an informative combination of features in different areas of the search space. Wherein on parallel nodes of the computing system, it is suggested to perform the most resource-intensive operations related to the estimation of the current set of agents, as well as the need to create and modify new sets of solutions based on stochastic computations. This allows to speedup the process of multi-agent search for an informative combination of features, and also decreases the practical threshold for applying the multi-agent method with indirect communication between agents to reduce big amounts of data.

The practical value of the work lies in the fact that the program implementation of the proposed method for the CPU cluster and for the GPU has been developed. It allows to perform feature selection in a parallel computer system for significantly less time compared to other feature selection methods, implemented, as a rule, sequentially.

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. Salfner F. A survey of online failure prediction methods / F. Salfner, M. Lenk, M. Malek // ACM computing surveys. –2010. – Vol. 42, Issue 3. – P. 1–42. DOI: 10.1145/ 1670679.1670680.
2. Bezdek J. C. Pattern Recognition with Fuzzy Objective Function Algorithms / J. C. Bezdek. – N.Y. : Plenum Press, 1981. – 272 p. DOI: 10.1007/978-1-4757-0450-1.
3. Bow S. Pattern recognition and image preprocessing / S. Bow. – New York: Marcel Dekker Inc., 2002. – 698 p. DOI: 10.1201/9780203903896.
4. Shin Y. C. Intelligent systems : modeling, optimization, and control / C. Y. Shin, C. Xu. – Boca Raton: CRC Press, 2009. – 456 p. DOI: 10.1201/9781420051773.
5. Bishop C. M. Pattern recognition and machine learning / C. M. Bishop. – New York : Springer, 2006. – 738 p.
6. Encyclopedia of machine learning / [eds. C. Sammut, G.I. Webb]. – New York : Springer, 2011. – 1031 p. DOI: 10.1007/978-0-387-30164-8.
7. Abonyi J. Cluster analysis for data mining and system identification / J. Abonyi, B. Feil. – Basel: Birkhäuser, 2007. – 303 p.
8. Jensen R. Computational intelligence and feature selection: rough and fuzzy approaches / R. Jensen, Q. Shen. – Hoboken: John Wiley & Sons, 2008. – 339 p. DOI: 10.1002/9780470377888.
9. Lee J. A. Nonlinear dimensionality reduction / J. A. Lee, M. Verleysen. – New York : Springer, 2007. – 308 p. DOI: 10.1007/ 978-0-387-39351-3.
10. Bodyanskiy Ye. A Multidimensional Cascade Neuro-Fuzzy System with Neuron Pool Optimization in Each Cascade / Ye. Bodyanskiy, O. Tyshchenko, D. Kopaliani // Int. Journal of Information Technology and Computer Science (IJITCS). – 2014. – Vol. 6, No. 8. – P. 11–17. DOI: 10.5815/ijitcs.2014.08.02
11. Oliinyk A. Production rules extraction based on negative selection / A. Oliinyk // Radio Electronics, Computer Science, Control. – 2016. – № 1. – P. 40–49. DOI: 10.15588/1607-3274-2016-1-5.
12. 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.
13. 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.
14. Oliinyk A. O. Agent technologies for feature selection / A. O. Oliinyk, O. O. Oliinyk and S. A. Subbotin // Cybernetics and Systems Analysis. – 2012. – Vol. 48, Issue 2. – P. 257–267. DOI: 10.1007/s10559-012-9405-z.
15. Jolliffe I. T. Principal Component Analysis / I. T. Jolliffe. – Berlin : Springer-Verlag. – 2002. – 489 p.
16. McLachlan G. Discriminant Analysis and Statistical Pattern Recognition / G. McLachlan. – New Jersey : John Wiley & Sons. – 2004. – 526 p.
17. Guyon I. An introduction to variable and feature selection / I. Guyon, A. Elisseeff // Journal of machine learning research. – 2003. – № 3. – P. 1157–1182.
18. Kim D. H. Bacterial Foraging Based Neural Network Fuzzy Learning / D. H. Kim, C. H. Cho // Proceedings of the 2nd Indian International Conference on Artificial Intelligence (IICAI-2005). – Pune : IICAI, 2005. – P. 2030–2036.
19. Субботін С. О. Неітеративні, еволюційні та мультиагентні методи синтезу нечіткологічних і нейромережних моделей: монографія / С. О. Субботін, А. О. Олійник, О. О. Олійник ; під заг. ред. С. О. Субботіна. – Запоріжжя : ЗНТУ, 2009. – 375 с.
20. 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.

Article was submitted 29.05.2017.

Олійник А. О.[1], Скрупський С. Ю.[2], Шкарупило В. В.[3], Благодарьов О. Ю.[4]

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

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

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

[4]Аспірант кафедри програмних засобів, Запорізький національний технічний університет, Запоріжжя, Україна

ПАРАЛЕЛЬНИЙ МУЛЬТИАГЕНТНИЙ МЕТОД РЕДУКЦІЇ ВЕЛИКИХ МАСИВІВ ДАНИХ ДЛЯ РОЗПІЗНАВАННЯ ОБРАЗІВ

Актуальність. Вирішено задачу відбору інформативних ознак при обробці великих масивів даних на основі мультиагентного підходу та паралельних обчислень. Об'єкт дослідження – процес відбору інформативних ознак. Предмет дослідження – методи відбору інформативних ознак.

Мета роботи полягає в створенні паралельного мультиагентного методу редукції великих масивів даних.

Метод. Запропоновано паралельний мультиагентний метод редукції великих масивів даних. Розроблений метод передбачає розбиття множини агентів на декілька підмножин для паралельного пошуку інформативної комбінації ознак в різних областях простору пошуку. При цьому на паралельних вузлах обчислювальної системи запропоновано виконувати найбільш ресурсомісткі операції, пов'язані з оцінюванням поточної множини агентів, а також з необхідністю створення і модифікації нових множин рішень на основі стохастичних обчислень. Це дозволяє прискорити процес мультиагентного пошуку інформативної комбінації ознак, а також знизити практичний поріг застосування мультиагентного методу з непрямим зв'язком між агентами для редукції великих масивів даних.

Результати. Розроблено програмне забезпечення, яке реалізує запропонований метод і дозволяє виконувати відбір інформативних ознак на основі мультиагентного підходу і паралельних обчислень.

Висновки. Проведені експерименти підтвердили працездатність запропонованого математичного забезпечення та дозволяють рекомендувати його для використання на практиці при обробці великих масивів даних для розпізнавання образів. Перспективи подальших досліджень можуть полягати в модифікації розробленого методу шляхом використання різних критеріїв оцінювання групової інформативності ознак, а також експериментальному дослідженні запропонованого методу на більшому комплексі практичних завдань різної природи і розмірності.

Ключові слова: агент, вибірка даних, відбір ознак, паралельні обчислення, мультиагентний підхід, розпізнавання образів.

Олейник А. А.[1], Скрупский С. Ю.[2], Шкарупило В. В.[3], Благодарев А. Ю.[4]

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

[4]Аспирант кафедры программных средств, Запорожский национальный технический университет, Запорожье, Украина

ПАРАЛЛЕЛЬНЫЙ МУЛЬТИАГЕНТНЫЙ МЕТОД РЕДУКЦИИ БОЛЬШИХ МАССИВОВ ДАННЫХ ДЛЯ РАСПОЗНАВАНИЯ ОБРАЗОВ

Актуальность. Решена задача отбора информативных признаков при обработке больших массивов данных на основе мультиагентного подхода и параллельных вычислений. Объект исследования – процесс отбора информативных признаков. Предмет исследования – методы отбора информативных признаков.

Цель работы заключается в создании параллельного мультиагентного метода редукции больших массивов данных.

Метод. Предложен параллельный мультиагентный метод редукции больших массивов данных. Разработанный метод предполагает разбиение множества агентов на несколько подмножеств для параллельного поиска информативной комбинации признаков в различных областях пространства поиска. При этом на параллельных узлах вычислительной системы предложено выполнять наиболее ресурсоемкие операции, связанные с оцениванием текущего множества агентов, а также с необходимостью создания и модификации новых множеств решений на основе стохастических вычислений. Это позволяет ускорить процесс мультиагентного поиска информативной комбинации признаков, а также снизить практический порог применения мультиагентного метода с непрямой связью между агентами для редукции больших массивов данных

Результаты. Разработано программное обеспечение, которое реализует предложенный метод и позволяет выполнять отбор информативных признаков на основе мультиагентного подхода и параллельных вычислений.

Выводы. Проведенные эксперименты подтвердили работоспособность предложенного математического обеспечения и позволяют рекомендовать его для использования на практике при обработке больших массивов данных для распознавания образов. Перспективы дальнейших исследований могут заключаться в модификации разработанного метода путем использования различных критериев оценивания групповой информативности признаков, а также экспериментальном исследовании предложенного метода на большем комплексе практических задач разной природы и размерности.

Ключевые слова: агент, выборка данных, отбор признаков, параллельные вычисления, мультиагентный подход, распознавание образов.

REFERENCES

1. Salfner F., Lenk M., Malek M. A survey of online failure prediction methods, *ACM computing surveys*, 2010, Vol. 42, Issue 3, pp. 1–42. DOI: 10.1145/1670679.1670680.
2. Bezdek J. C. Pattern Recognition with Fuzzy Objective Function Algorithms. N.Y. Plenum Press, 1981, 272 p. DOI: 10.1007/978-1-4757-0450-1.
3. Bow S. Pattern recognition and image preprocessing. New York, Marcel Dekker Inc., 2002, 698 p. DOI: 10.1201/9780203903896.
4. Shin Y. C., Xu C. Intelligent systems : modeling, optimization, and control. Boca Raton: CRC Press, 2009, 456 p. DOI: 10.1201/9781420051773.
5. Bishop C. M. Pattern recognition and machine learning, New York, Springer, 2006, 738 p.
6. Sammut C., Webb G. I. eds. Encyclopedia of machine learning. New York, Springer, 2011, 1031 p. DOI: 10.1007/978-0-387-30164-8.
7. Abonyi J., Feil B. Cluster analysis for data mining and system identification. Basel, Birkhäuser, 2007, 303 p.
8. Jensen R., Shen Q. Computational intelligence and feature selection: rough and fuzzy approaches. Hoboken, John Wiley & Sons, 2008, 339 p. DOI: 10.1002/9780470377888.
9. Lee J. A., Verleysen M. Nonlinear dimensionality reduction. New York, Springer, 2007, 308 p. DOI: 10.1007/978-0-387-39351-3.
10. Bodyanskiy Ye., Tyshchenko O., Kopaliani D. A Multidimensional Cascade Neuro-Fuzzy System with Neuron Pool Optimization in Each Cascade, *Int. Journal of Information Technology and Computer Science (IJITCS),* 2014, Vol. 6, No. 8, pp. 11–17. DOI: 10.5815/ijitcs.2014.08.02
11. 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.
12. 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.
13. Oliinyk A., Subbotin S. A. Association Rules Extraction for Pattern Recognition, *Pattern Recognition and Image Analysis*, 2016, Vol. 26, No. 2, pp. 419–426.
14. Oliinyk A. O., Oliinyk O. O. and Subbotin S. A. Agent technologies for feature selection, *Cybernetics and Systems Analysis,* 2012, Vol. 48, Issue 2, pp. 257–267. DOI: 10.1007/s10559-012-9405-z.
15. Jolliffe I. T. Principal Component Analysis. Berlin, Springer-Verlag, 2002, 489 p.
16. McLachlan G. Discriminant Analysis and Statistical Pattern Recognition. New Jersey, John Wiley & Sons, 2004, 526 p.
17. Guyon I., Elisseeff A. An introduction to variable and feature selection, *Journal of machine learning research*, 2003, No. 3, pp. 1157–1182.
18. Kim D. H., Cho C. H. Bacterial Foraging Based Neural Network Fuzzy Learning, *Proceedings of the 2nd Indian International Conference on Artificial Intelligence (IICAI-2005)*. Pune, IICAI, 2005, pp. 2030–2036.
19. Subbotin S., Oliinyk A., Oliinyk O. Noniterative, evolutionary and multi-agent methods of fuzzy and neural network models synthesis : monograph. Zaporizhzhya, ZNTU, 2009, 375 p. (In Ukrainian).
20. 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.