UDC 004.272.26: 004.93

Oliinyk A.[1], Subbotin S.[2], Skrupsky S.[3], Lovkin V.[4], Zaiko T.[5]

[1]PhD., Associate Professor of Department of Software Tools, Zaporizhzhia National Technical University, Zaporizhzhia, Ukraine
[2]Dr.Sc, Head of Department of Software Tools, Zaporizhzhia National Technical University, Zaporizhzhia, Ukraine
[3]PhD, Associate Professor of Computer Systems and Networks Department, Zaporizhzhia National Technical University, Zaporizhzhia, Ukraine
[4]PhD, Associate Professor of Department of Software Tools, Zaporizhzhia National Technical University, Zaporizhzhia, Ukraine
[5]PhD., Senior Lecture of Department of Software Tools, Zaporizhzhia National Technical University, Zaporizhzhia, Ukraine

INFORMATION TECHNOLOGY OF DIAGNOSIS MODELS SYNTHESIS BASED ON PARALLEL COMPUTING

Context. The problem of diagnosis models synthesis in the big data processing based on parallel computing is solved. The object of the research is the process of diagnosis models synthesis. The subject of the research are the methods and information technologies for diagnosis models synthesis.

Objective. The research objective is to develop diagnosis models synthesis information technology.

Method. The paper deals with information technology of diagnosis models synthesis which is a set of diagrams graphically describing structural elements of the system as well as the behavioral aspects of their interaction at various stages of diagnostics objects models construction. The developed information technology enables to perform the construction of distributed diagnostics systems where computationally complex stages of diagnosis models synthesis are performed on high-performance server equipment, which makes it possible to significantly increase the practical threshold for using diagnostics systems in the processing of big data sets for solving of the tasks of training sample data reduction, rules extraction, diagnosis models construction and retraining.

Results. The software which implements the proposed information technology and allows to synthesize diagnosis models based on the given data samples has been developed.

Conclusions. The conducted experiments have confirmed the proposed information technology operability and allow to recommend it for solving the problems of big data processing for technical and biomedical diagnostics in practice. The prospects for further researches may include the modification of the developed information technology by introducing of other methods of diagnosis models synthesis.

Keywords: data sample, diagnosis, rule extraction, feature selection, parallel computing, model synthesis.

NOMENCLATURE

CPU – Central Processing Unit;

GPU – Graphical Processing Unit;

IGA – Island Genetic Algorithm;

UML – Unified Modeling Language;

E_{IGA} – efficiency of the computer system in the implementation of the IGA method;

E_p – efficiency of the computer system in the implementation of the proposed method;

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

$Overhead_{IGA}$ – communication and synchronization costs of the IGA method;

$Overhead_p$ – communication and synchronization costs of the proposed method;

P – set of features (attributes) of observations in the given sample;

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

S – sample of observations (training sample);

S_{IGA} – speedup of the IGA method;

S_p – speedup of the proposed method;

T – set of output parameter values;

T_{IGA} – execution time of the IGA method;

T_p – execution time of the proposed method.

INTRODUCTION

Solving of technical and medical diagnostics tasks, as well as the task of nondestructive product quality control, is connected with the task of diagnosis models construction [1–6]. Such models enable to classify samples which are diagnosed with high accuracy, and at that the target classes are sufficiently convenient for perception and analysis by experts in the application areas.

Methods which enable to solve training sample data reduction, rules extraction, diagnosis models detection and retraining problems are presented in the papers [7–16]. However such methods based on sequential computing require significant time costs, which makes it difficult to use such methods for diagnostic decision making process automation in practice. So it is necessary to develop information technology which is capable to accelerate diagnosis models synthesis process based on parallel computing [17–20].

The research objective is to develop diagnosis models synthesis information technology which enables to build distributed diagnostics systems where computationally complex stages of model synthesis are performed on high-performance server equipment, which makes it possible to significantly increase the practical threshold for using diagnostic systems in big data sets processing.

1 PROBLEM STATEMENT

Suppose we have data sample $S = <P, T>$, which consists of Q samples. Every sample is characterized by values of attributes p_{q1}, p_{q2}, …, p_{qM} and output parameter t_q, where p_{qm} is a value of the m-th attribute of the q-th sample, $q = 1, 2, ..., Q$, $m = 1, 2, ..., M$; M – overall number of attributes in the set S. Then the problem of diagnosis models synthesis is to define model DM for

the sample $S = <P,T>$, which allows to approximate the set S with acceptable error level E. Error E [2, 21–23] of recognition using synthesized model DM can be evaluated as ratio of incorrectly recognized samples number N_{er} to overall samples number Q in the sample $S = <P,T>$:

$$E = \frac{N_{er}}{Q}.$$

2 REVIEW OF THE LITERATURE

As stated above, in the papers [7–16] methods, which enable to solve problems concerning diagnosis models synthesis, are proposed. However such methods based on sequential computing require significant time costs which makes it difficult to use such methods for diagnostic decision making process automation in practice.

Feature selection methods, which were proposed in the papers [7–9], enable to select informative feature combinations based on evolutionary [24–26] and multi-agent technologies [27] of computational intelligence. The proposed methods use aprioristic information about individual self-descriptiveness, reducing search space and decreasing time of informative feature combination selection. Nevertheless such methods require significant time costs for implementation during processing of high dimensionality data.

Productional rules extraction methods [10–12] enable to find the most significant replications $X \rightarrow Y$ from the given data samples $S = <P,T>$. It provides improvement of cumulative properties of diagnosis models which are synthesized, as well as increases interoperability of models, decreases its dimensionality (structural and parametrical complexity), utilized storage capacity and response speed.

The method of parametrical identification of neuro-fuzzy networks based on parallel random search, which was proposed in [13–16], uses stochastic optimization for synthesized models parameters setting (parameters of membership functions and weight coefficients of neuroelements), forms initial solution set subject to training sample information (significance of feature terms according to the compactness of training set samples arrangement in the corresponding term and the degree of its effect on output parameter value). It makes it possible to move initial search points near optimal values and to accelerate optimization process.

It is necessary to develop information technology which enables to use the methods proposed in the papers [7–16] in practice for construction of distributed diagnostics systems where computationally complex stages of diagnosis models synthesis are performed on high-performance server equipment, significantly increasing the practical threshold for using diagnostic systems.

3 MATERIALS AND METHODS

As it was mentioned above, diagnosis models construction based on available data sets generally requires considerable computational resources. Therefore the developed information technology is implemented based on «client-server» architecture [28–30]. At that it is proposed to implement complex computational processes concerning training sample processing, big data storing, model

synthesis etc., on server side, organizing client access according to their access permissions. Clients are understood as people and computer systems which solve practical tasks concerning construction and application of diagnosis models.

The diagram describing main system functions of information technology front-end and back-end is presented in Fig. 1.

As it is described in Fig. 1, main functions of information system are divided between server and client sides. Client side of information system provides implementation of user interface, processing of data, entered by user (verification of data format correctness), and calculation of diagnostics object output parameter value based on the synthesized model. Server side of the system should use high-performance equipment for solving of the tasks concerning diagnosis models construction. Computationally complex processes concerning reduction of training sample dimensionality, rules extraction, synthesis and retraining of diagnosis models are processed on the server side. Besides it is proposed to use database for storage and processing of training samples from different users and also of synthesized models, extracted rules, reduced samples and other results of system operation.

For the design and development possibility of diagnosis models synthesis software system, corresponding information technology presented as the set of UML-models [31–33] is proposed. UML-models are represented as diagrams (Fig. 2–7), graphically describing structural and behavioral aspects of construction of distributed diagnostics systems, which enable to solve the tasks of training sample data reduction, rules extraction, diagnosis models construction and retraining.

Based on the main functions of the system (Fig. 1) and also on the chosen architecture of diagnosis models synthesis information technology, it is possible to define general configuration and topology of the system as the model presented in deployment diagram (Fig. 2).

System consists of three nodes: server, client computer and database server. Software implementation of intellectual methods which are used at various stages of diagnosis models synthesis (data reduction, rules extraction, model construction, model retraining) is contained on the server. Set of client computers with installed software can interact with the server. Work of database server is organized through interaction between database management system and database. Interaction of user (client computers) with database is performed indirectly through the server, where user access permissions for corresponding data sets are verified.

For representation of different user interactions with the system the model was figured as use case diagram (Fig. 3).

As can be seen from the Fig. 3, there are two user kinds which interact with the system: user and administrator. Interaction between them is realized in the use case "Registration", when administrator approves user permissions for access to the other system functions.

System user is a person which uses information technology for solving practical diagnostics tasks. User should be registered in the system to provide access of
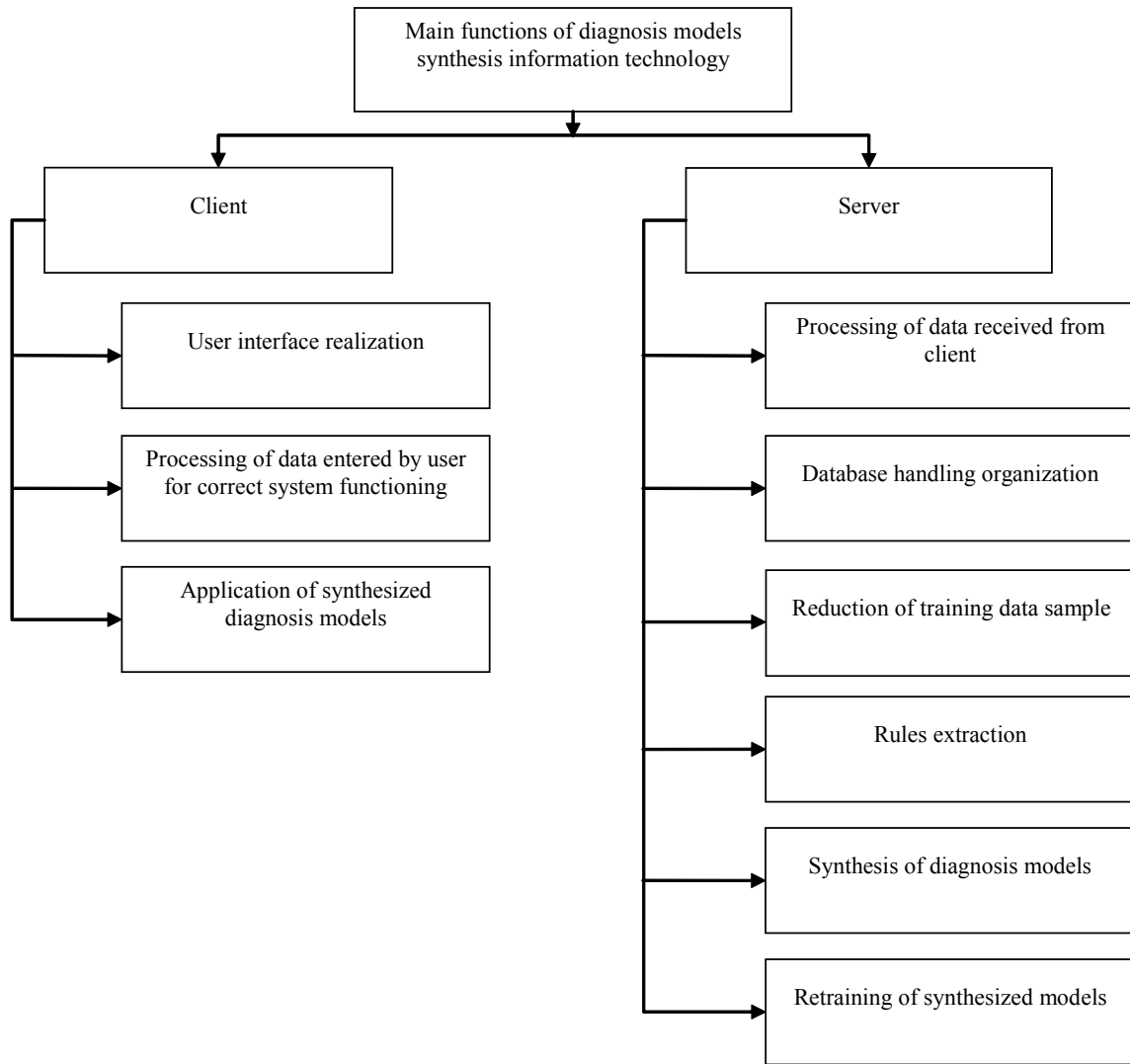
Figure 1 – Main functions of diagnosis models synthesis information technology
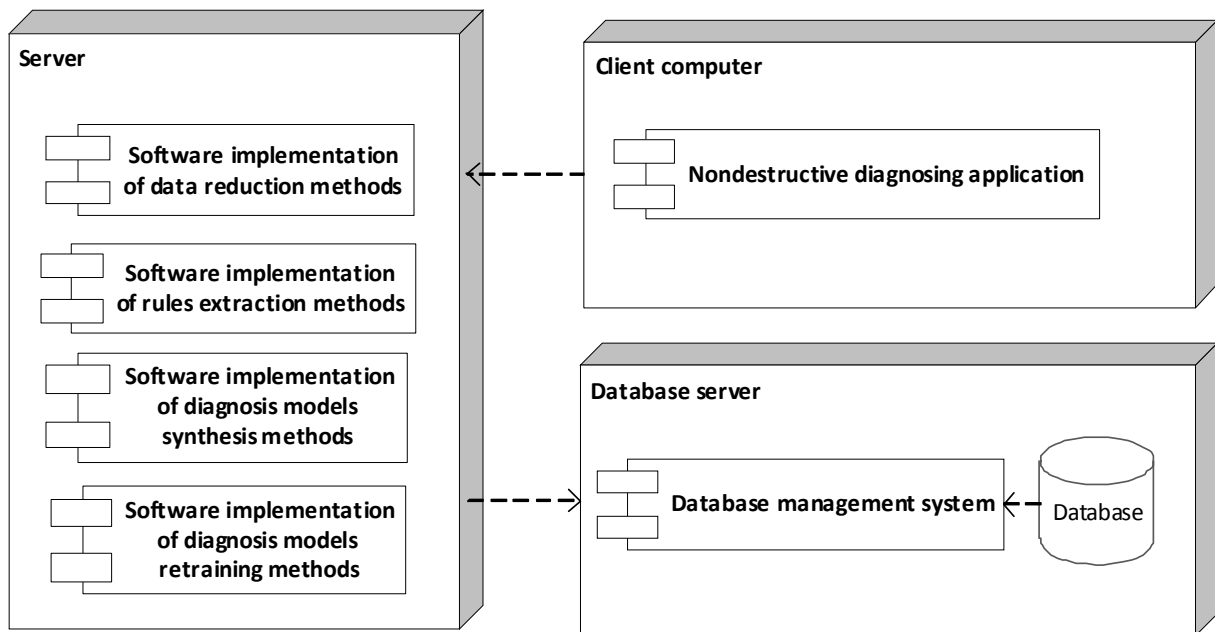

Figure 2 – Deployment diagram of diagnosis models synthesis information technology (UML 2.5 notation)

Removal of irrelevant data from database

Administrator

«extends»

User registration cancellation

User registration confirmation

Registration

Input and editing of learning sample data

User

Sample reduction

Rules extraction

Diagnosis models synthesis

Model retraining

Diagnosing based on synthesized model

Figure 3 – Use case diagram of diagnosis models synthesis information technology

different users to the corresponding data samples. User can interact with the system during registration, input and editing of training sample, solving of diagnostics tasks (training sample dimensionality reduction, extraction of new knowledge about the dependencies which are researched, synthesis or diagnostics model retraining, diagnostics based on the synthesized model).

System administrator provides user registration through input and editing of information about users and user registration confirmation, and also provides access to the database for removal or archiving of information which is not used during long period of time and reduction in such a way of physical volume of the disk space which is in use.

With the object of modeling of logical operation and actions performed in the diagnosis models synthesis information system, the corresponding model was created and is presented in the Fig. 4 as activity diagram.

As it is presented in the Fig. 4, user enters account data at the beginning of system usage. After that system proposes to choose one of the following operating modes:

– diagnosis model synthesis – computationally complex process, which is realized on server, performing stages of training sample data reduction, rules extraction, diagnosis model construction;

– retraining of the synthesized model;

– diagnosing using the synthesized model – calculation of diagnostics object output parameter value using the model based on the measured input parameters.

Then user can continue to use the system or can quite.

The model of distribution of interaction between information system objects and users in time is represented as sequence diagram (Fig. 5–7). In the diagram the process of interaction between system components through calls of procedures, which realize corresponding use cases, is presented.

In the sequence diagram, presented in the Fig. 5, system use cases concerning user registration, input, editing of training sample and removing of irrelevant data (Fig. 3) are represented. As can be seen, not only users but also system administrator takes part in these processes.

Processes connected with preliminary processing of training sample for diagnosis model synthesis (reduction of data sample and rules extraction) are performed on user request (through client computer) on the server, and the results are saved in the database (Fig. 6). Processes, presented in the sequence diagram (Fig. 6), correspond to the system use cases "Sample reduction" and "Rules extraction" (Fig. 3), which are performed by user without administrator participation.
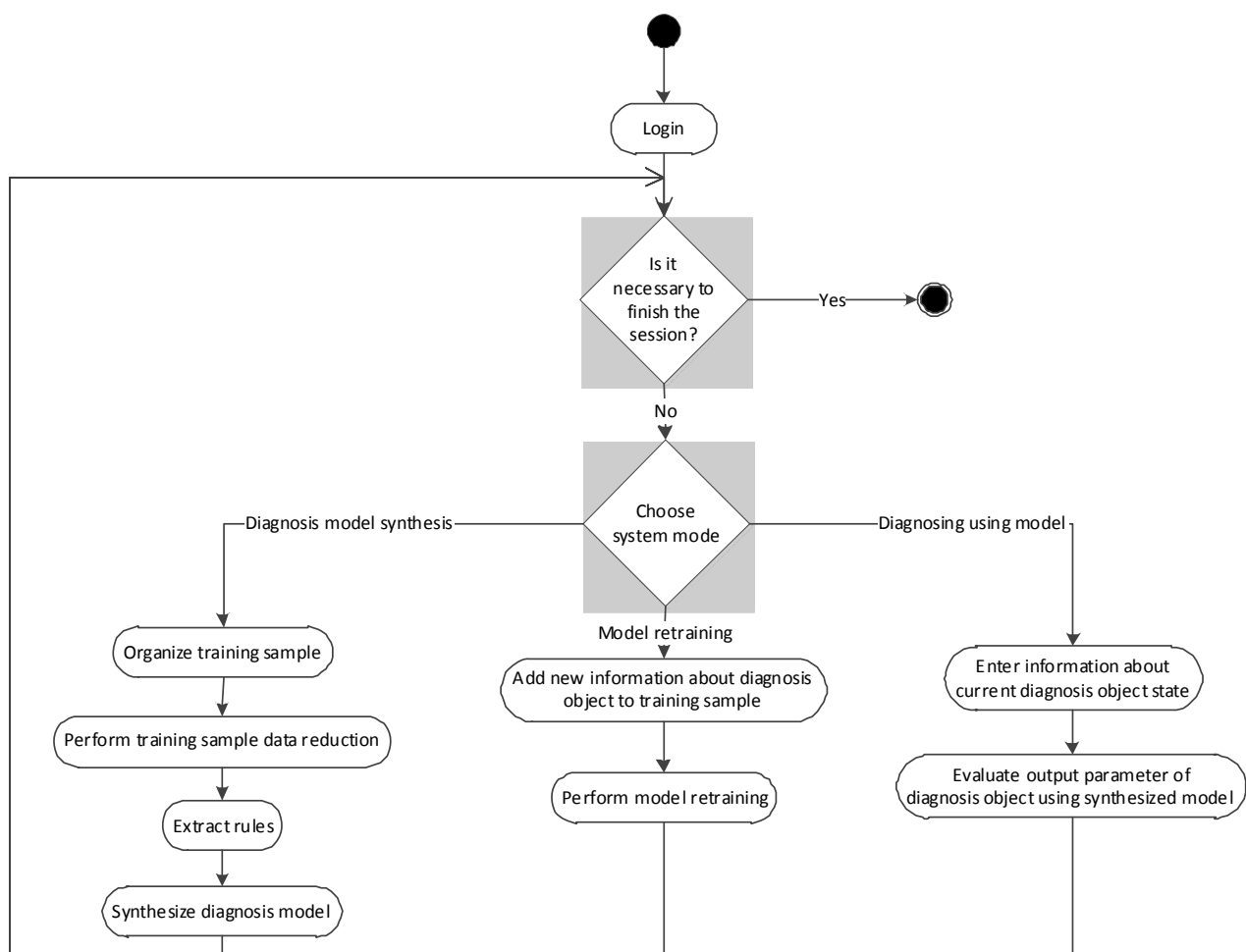

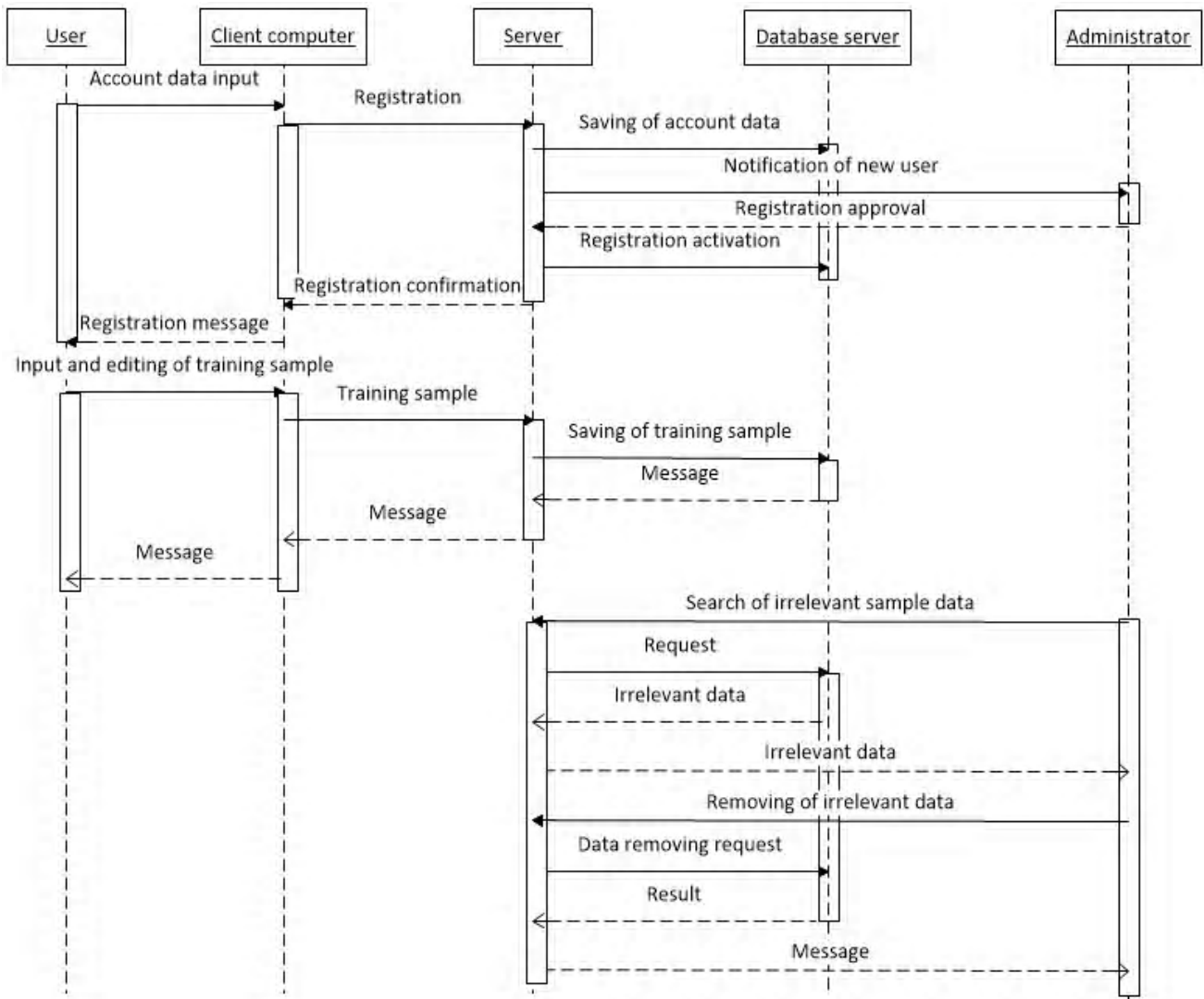Figure 4 – Activity diagram of diagnosis models synthesis information technology

Figure 5 – Sequence diagram of user registration and training sample processing

Computationally complex processes of model synthesis and retraining are also performed on the server on user request (Fig. 7) with saving of results to the database. The result of these processes (synthesized diagnosis model) is transmitted to client computer. It enables further usage of the synthesized model for diagnosis object or process output parameter value calculation. Such approach provides diagnostics process execution on client computer without server access.

Thus the developed information technology of diagnosis models synthesis is represented by the set of diagrams (Fig. 3–8), which graphically describe structural elements of the system, and also behavioral aspects of its interaction at various stages of diagnostics objects models construction. The proposed information technology allows to construct distributed diagnostics systems, where computationally complex stages of diagnosis models synthesis are performed on the high-performance server equipment, which enables to significantly increase the practical threshold for using diagnostic systems which are capable of solving the tasks of training sample data reduction, rules extraction, diagnosis models construction and retraining.

As it was mentioned above, it is supposed to use database for storage and processing of information about objects or processes which are researched (training samples) and also for system operation results (synthesized models, extracted rules, reduced samples etc.) in the developed diagnosis models synthesis information technology. Database contains set of tables, which are connected in some way and contain information about users and samples of data representing objects and processes which are researched [34–36]. ER-model of the developed diagnosis models synthesis information technology database supposes availability of the following entities:

– Users – contains information about system users. Entity fields: id – user number (unique identifier); userLogin – login of user; userPass – password of user; info – information about user;

– Groups – reflects user groups and group data. Fields: id – primary key, name – group name, info – group description;

– UserGroups – describes correspondence between users and their groups. At that idGroup, idUser are foreign keys pointing at the corresponding entities;
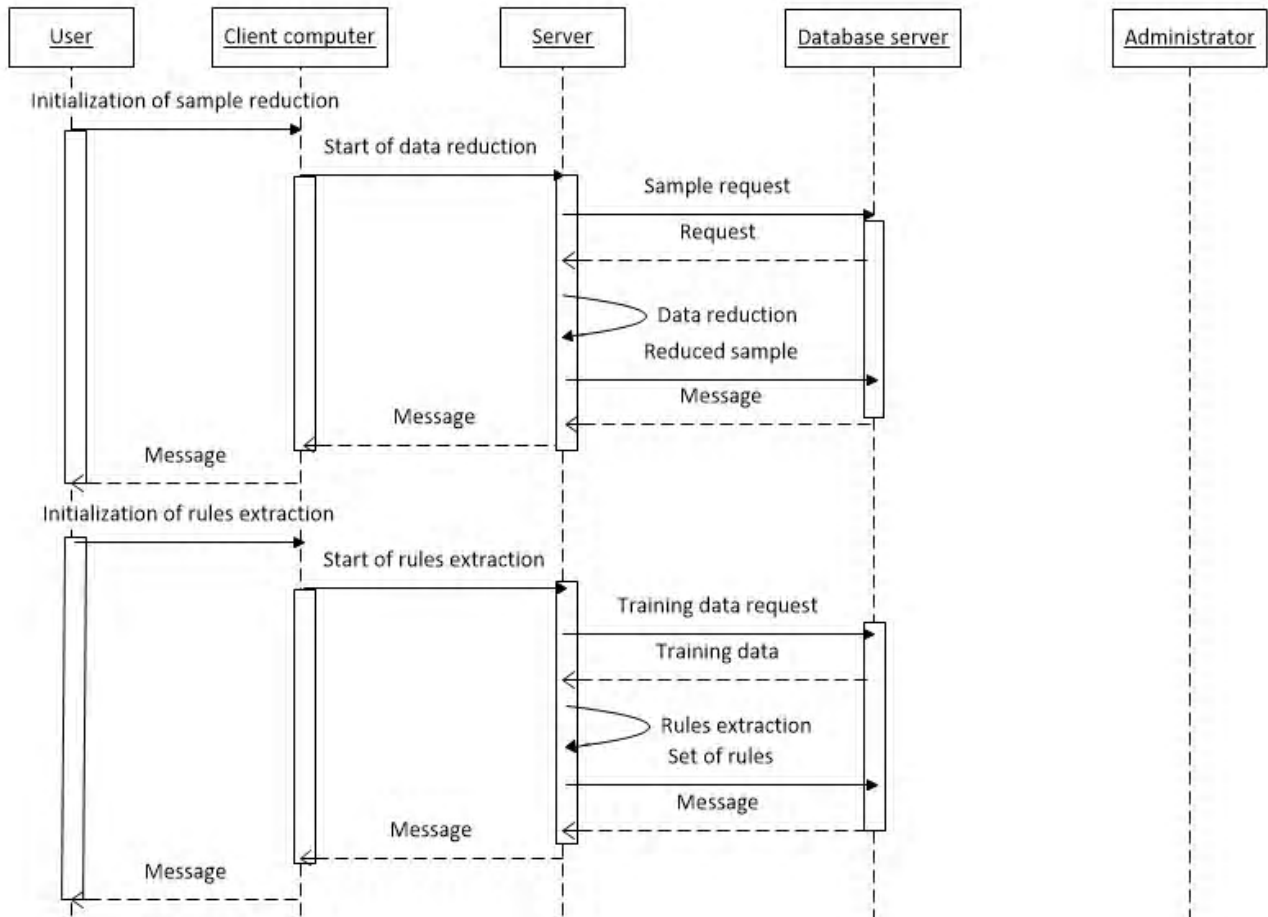
Figure 6 – Sequence diagram of training sample reduction and rules extraction

– Samples – contains information about training samples. Entity fields: id – unique identifier of the corresponding data sample; ref – reference to the file with the table which contains training sample $S = <P,T>$ with the identifier id; info – description of training sample; idGroups – reference to the user group which has permissions for access to the sample id;

– SamplesRed – contains information about the samples which were reduced using the methods implemented in the system through the reduction of training samples from the entity Samples. Fields: id – unique identifier of reduced data sample; ref – reference to the file with the table which contains reduced sample, corresponding to the identifier id; info – description of reduced sample (particularly reduction method); idSample – identifier (foreign key) of the training sample which was the base for getting of the reduced sample;

– Rules – contains information about productional rules, which were extracted using methods implemented in the system through the processing of training samples from the entity Samples. Entity fields: id – unique identifier of rules set; ref – reference to the file which contains set of rules $P \to T$, corresponding to the identifier id; info – description of rules set (particularly rules extraction method); idSample – identifier (foreign key) of the training sample $S = <P,T>$, which was the base for getting of the set of rules $P \to T$;

– Models – contains information about diagnosis models, which were synthesized based on the given (idSample) or reduced (idSampleRed) samples, and also on the extracted rules (idRules) using the methods which were implemented in the system. Fields: id – unique identifier of diagnosis model; ref – reference to the file which contains structure and parameters of the model with identifier id; info – description of the synthesized model (for example, synthesis method); idSample, idSampleRed, idRules – identifiers (foreign keys) of the training sample, reduced sample and rules set correspondingly, which were the base for the synthesis of the model id. At the same time if only one data set is used in the synthesis of the model id, it is necessary to use reference to the dummy record in the other entities Samples, SamplesRed, Rules as values of the other foreign keys; idParentModel – model which was used as the base for synthesis of the model id (this parameter is not required, because it is necessary only if synthesized model is retrained); idGroups – reference to the group of users having access to the implementation of the synthesized model.

Database scheme is presented in the Fig. 8.

As it is presented in the Fig. 8, connections between entities are created from foreign to primary key, and integrity control is provided by database management system tools.

The developed database supports storage and processing of information about objects and processes which are researched (training samples) and also system
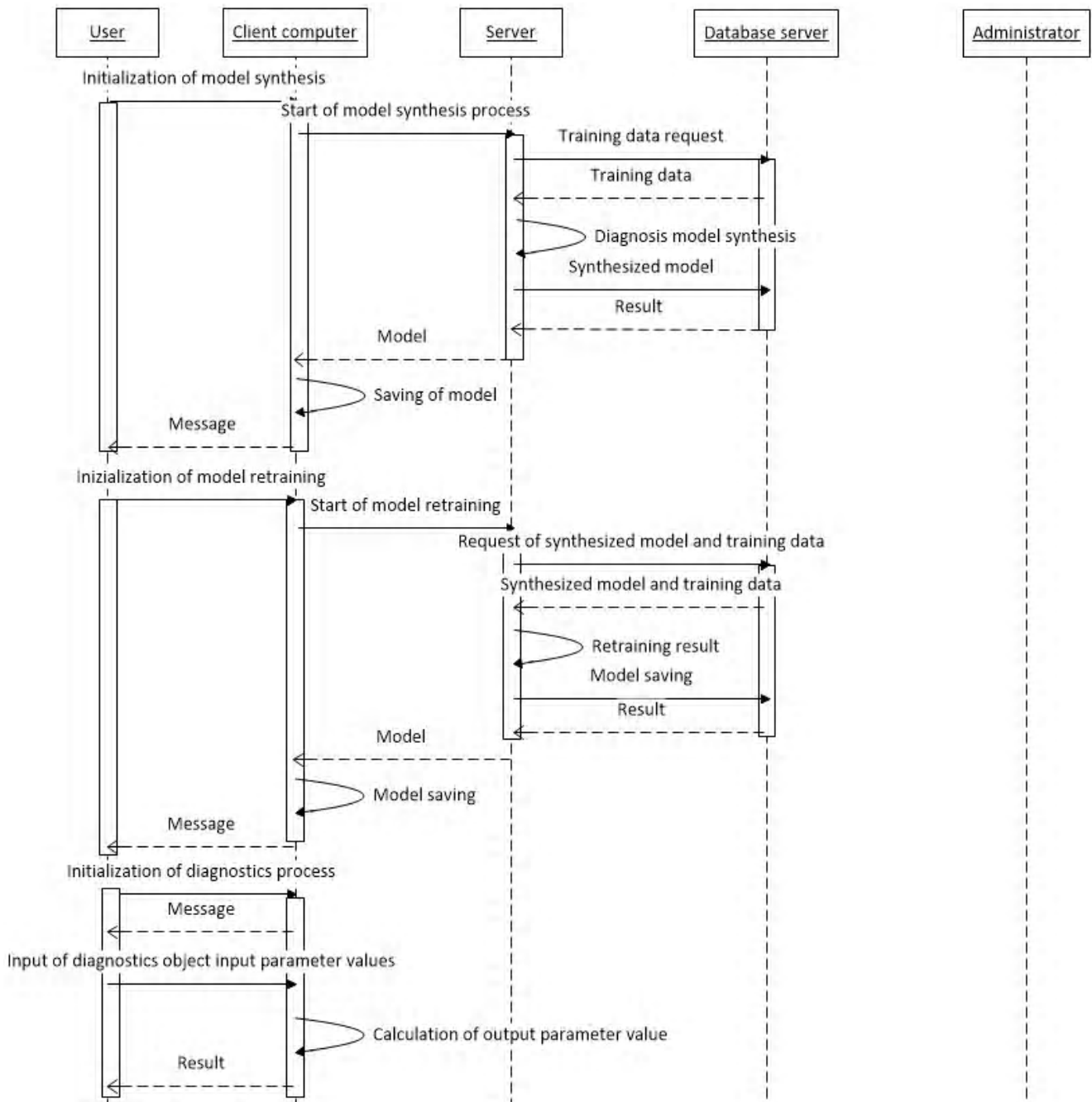
Figure 7 – Sequence diagram of synthesis and usage of diagnosis models

operation results (structure and parameters of synthesized models, extracted rules, reduced samples etc.) in the developed diagnosis models synthesis information technology.

4 EXPERIMENTS

For efficiency examination of the proposed information technology corresponding software system for diagnosis models synthesis was developed.

The software was developed based on Java programming language and architecture pattern MVC. Graphic part (View) was implemented using SWING package. The application has the following class structure. Class Model represents classes which describe entities of database

ER-model (Fig. 8). Fields of these classes are identical with fields of database tables. It enables to apply modern programming frameworks, for example, Hibernate. Every row of database table can be got as one class instance in the application. At that several table rows form collection. Classes View are frames (forms, dialog boxes), which contain graphical user interface and allow user to interact with the application. Classes Controller connect to the server, realize event service and handling of user actions. Class LoginController enables to connect to the server using login and password or to register (to create user account which is inserted to the database). MainController enables to handle user instructions of calling appropriate forms. It sends instruction to the server and gets permission (according to
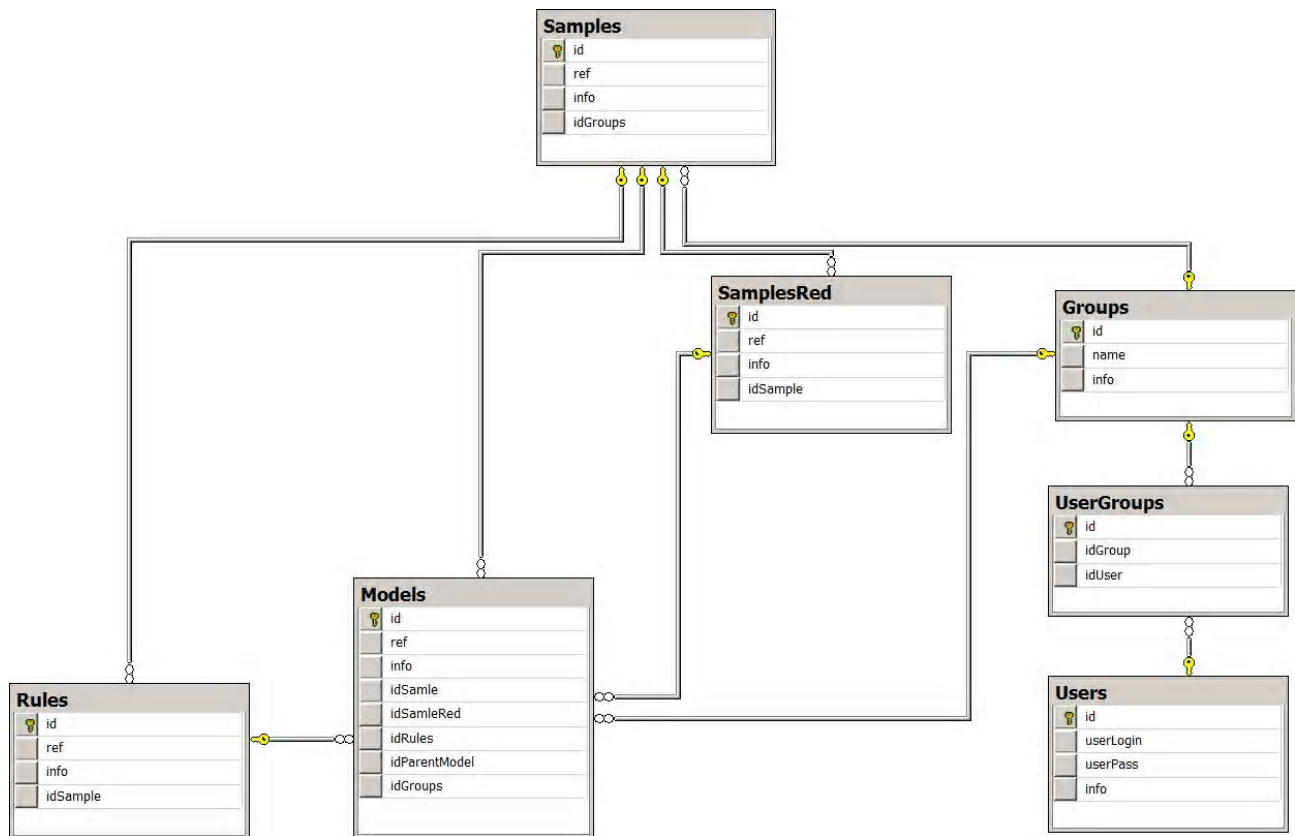
146

Figure 8 – Diagnosis models synthesis information technology database scheme

user access permissions) to call appropriate function, for example, diagnostics or synthesis. Class WorkWithSampleController gives methods, which enable to load sample from the file with the following sending of it to the server through ConnectionController. Besides it enables to edit sample and to reduce it. Class SynthesisController downloads samples and synthesis methods which are available for user from database and also performs models synthesis. Class AdditionalTrainingController gives methods, which get models and retraining methods available for user from database, besides it enables to extend sample and to retrain model. DiagnosisController enables to access diagnosis models, to input model parameters and to diagnose with the following saving of the result to the file.

Numerical experiments on the efficiency of the proposed information technology and the corresponding software system were performed by solving of different practical diagnostics problems [37–39], particularly the task of hypertensive patient health status prediction [37]. Applying

the developed mathematical (methods [7–16] and the proposed information technology) and software support the tasks of learning sample reduction (informative attributes identification and production rules extraction) and diagnosis models synthesis were solved sequentially. Numerical results of the developed information technology application for neuro-fuzzy diagnosis models definition using the proposed parallel method based on stochastic computation [24–27, 40] and island method of evolutionary search (Island Genetic Algorithm, IGA) [24–26] are presented. Experiments were executed on 1, 2, 4, 8 and 16 cores of CPU cluster [41] as well as on graphic engine GPU [42].

5 RESULTS

Experimental results using CPU cluster are presented in the table 1.

The results of experiments using graphic engine GPU NVIDIA GTX 285+, which was programmed based on CUDA technology [43], are presented in the table 2. Speedup of computational process was measured regarding one CPU.

Table 1 – The results of experiments using CPU cluster

CPU number	T_p, s	T_{IGA}, s	S_p	S_{IGA}	Overhead$_p$, s	Overhead$_{IGA}$, s	E_p	E_{IGA}
1	101.18	128.5	1	1	0	0	1	1
2	57.67	74.53	1.75	1.72	8.07	11.92	0.88	0.86
4	30.61	39.19	3.31	3.28	6.43	8.62	0.83	0.82
8	18.09	23.61	5.59	5.44	7.78	11.1	0.7	0.68
16	11.45	15.34	8.84	8.38	9.27	13.96	0.55	0.52

Table 2 – The results of experiments realized on GPU

Number of GPU threads	T_p, s	T_{IGA}, s	S_p	S_{IGA}	Overhead$_p$, s	Overhead$_{IGA}$, s
60	39.8	60.22	2.54	2.13	4.46	7.05
80	33.49	50.86	3.02	2.53	4.09	6.41
100	30.33	46.18	3.34	2.78	4.0	6.23
120	28.89	44.25	3.5	2.9	4.13	6.64
140	28.55	44.7	3.54	2.87	4.37	6.93
160	29.05	45.72	3.48	2.81	4.74	7.73
180	30.29	47.96	3.34	2.68	5.24	8.63
200	31.0	48.92	3.26	2.63	5.7	9.15
240	31.26	51.37	3.24	2.5	6.56	13.36

6 DISCUSSION

As it is presented in the tables 1 and 2, the proposed technology of diagnosis models synthesis allows to synthesize models with productivity similar to the models described in [3, 10, 13]. Thus, the method proposed in [13] due to application of new solution search operators modifications decreases number of processor operations, including communicatory costs, and so random search is realized quicker than in the IGA method [24–26] (for example, time of model synthesis for 16 cores of CPU equaled 11.45 s for the proposed method and 15.34 s for IGA method). At that the proposed diagnosis models synthesis technology provides construction of neuro-fuzzy models with acceptable accuracy. That is productivity rise is not provided due to decrease of diagnosis models approximating and resumptive properties level.

The efficiency of CPU cluster, which was used by the method proposed in [13] and the IGA method, is acceptable (particularly parallel system efficiency reaches 0.7 for the proposed method and 0.68 for the IGA method using 8 cores of CPU). Application of more than 8 cores of CPU isn't justified, because it greatly decreases system efficiency due to transmission and synchronization. If number of GPU threads rises above 140, speedup of computing process will decrease, because overheads considerably rise and at the same time threads begin to stand.

Thus diagnosis models synthesis technology which was proposed in the paper allows to efficiently apply modern parallel computing architectures for getting the result with appropriate accuracy in acceptable time. Usage of cross-platform language considerably extends scope of the proposed technology.

CONCLUSIONS

In this paper actual problem of diagnosis models synthesis process automation was solved.

Scientific novelty of the paper is in the proposed information technology of diagnosis models synthesis, which consists of the set of methods and diagrams which connect methods with each other, graphically describe structural elements of diagnostics systems and also behavioral aspects of its communication at various stages of diagnostics objects models construction. The proposed information technology enables to construct distributed diagnostics systems where computationally complex stages of diagnosis model synthesis are performed on high-performance server equipment, which makes it possible to significantly increase the practical threshold for using diagnostic systems in big data sets processing, solving the tasks of training sample data reduction, rules extraction, building and retraining of diagnosis models.

Practical significance of the paper consists in the solution of practical problems. Experimental results showed that the proposed information technology allowed to significantly increase the speed of diagnosis models synthesis process and it could be used in practice for solving of practical tasks concerning diagnostics and nondestructive product quality control.

ACKNOWLEDGMENTS

REFERENCES

1. Price C. Computer based diagnostic systems / C. Price. – London : Springer, 1999. – 136 p. DOI: 10.1007/978-1-4471-0535-0.
2. Bow S. Pattern recognition and image preprocessing / S. Bow. – New York : Marcel Dekker Inc., 2002. – 698 p. DOI: 10.1201/9780203903896.
3. Encyclopedia of machine learning / [eds. C. Sammut, G. I. Webb]. – New York: Springer, 2011. – 1031 p. DOI: 10.1007/978-0-387-30164-8.
4. 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.
5. 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.
6. 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.
7. Subbotin S. The Dimensionality Reduction Methods Based on Computational Intelligence in Problems of Object Classification and Diagnosis / S. Subbotin, A. Oliinyk // Recent Advances in Systems, Control and Information Technology. Advances in Intelligent Systems and Computing. – 2017. – Vol. 543. – P. 11–19. DOI: 10.1007/978-3-319-48923-0_2.
8. Subbotin S. The Sample and Instance Selection for Data Dimensionality Reduction / S. Subbotin, A. Oliinyk // Recent Advances in Systems, Control and Information Technology. Advances in Intelligent Systems and Computing. – 2017. – Vol. 543. – P. 97–103. DOI: 10.1007/978-3-319-48923-0_13.

9. Oliinyk A. A. Parallel multiagent method of big data reduction for pattern recognition / A. A. Oliinyk, S. Yu. Skrupsky, V. V. Shkarupylo, O. Blagodariov // Radio Electronics, Computer Science, Control. – 2017. – № 2. – C. 82–92.

10. 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.

11. 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.

12. Oliinyk A. A stochastic approach for association rule extraction / A. Oliinyk, S. A. Subbotin // Pattern Recognition and Image Analysis. – 2016. – Vol. 26, № 2. – P. 419–426. DOI: 10.1134/S1054661816020139.

13. Oliinyk A. O. Using Parallel Random Search to Train Fuzzy Neural Networks / A. O. Oliinyk, S. Yu. Skrupsky, S. A. Subbotin // Automatic Control and Computer Sciences. – 2014. – Vol. 48, Issue 6. – P. 313–323. DOI: 10.3103/S0146411614060078.

14. Oliinyk A. Parallel computing system resources planning for neuro-fuzzy models synthesis and big data processing / A. Oliinyk, S. Skrupsky, S. Subbotin, O. Blagodariov, Ye. Gofman // Radio Electronics, Computer Science, Control. – 2016. – № 4. – P. 61–69. DOI: 10.15588/1607-3274-2016-4-8.

15. The model for estimation of computer system used resources while extracting production rules based on parallel computations / [A. A. Oliinyk, S. Yu. Skrupsky, V. V. Shkarupylo, S. A. Subbotin] / / Radio Electronics, Computer Science, Control. – 2017. – № 1. – C. 142–152. DOI: 10.15588/1607-3274-2017-1-16.

16. Oliinyk A. Parallel Computer System Resource Planning for Synthesis of Neuro-Fuzzy Networks / A. Oliinyk, S. Skrupsky, S. Subbotin // Recent Advances in Systems, Control and Information Technology. Advances in Intelligent Systems and Computing. – 2017. – Vol. 543. – P. 88–96. DOI: 10.1007/978-3-319-48923-0_12.

17. David Kirk Programming Massively Parallel Processors 3rd Edition. A Hands-on Approach / Kirk David, Hwu Wen-mei, 2016. – 576 p. ISBN: 9780128119860.

18. A comparison of approaches to large-scale data analysis / [Andrew Pavlo, Erik Paulson, Alexander Rasint et all] // International Conference on Management of Data. – 2009. – P. 165–178. DOI: 10.1145/1559845.1559865

19. Luiz Andre Barroso The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines / Luiz Andre Barroso, Urs Hoelzle // Synthesis Lectures on Computer Architecture. – 2009. – Volume 4, Issue 1. – P. 154. DOI: 10.2200/S00193ED1V01Y200905CAC006

20. Zaigham Mahmood Data Science and Big Data Computing: Frameworks and Methodologies / Zaigham Mahmood // Springer International Publishing. – 2016. – P. 332. DOI: 10.1007/978-3-319-31861-5

21. 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.

22. 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.

23. Bishop C. M. Pattern recognition and machine learning / C. M. Bishop. – New York : Springer, 2006. – 738 p.

24. Gen M. Genetic algorithms and engineering design / M. Gen, R. Cheng. – New Jersey : John Wiley & Sons, 1997. – 352 p. DOI: 10.1002/9780470172254.

25. Yu X. Introduction to Evolutionary Algorithms (Decision Engineering) / X. Yu, M. Gen. – London : Springer, 2010. – 418 p. DOI: 10.1007/978-1-84996-129-5.

26. Ayala H. V. Cascaded evolutionary algorithm for nonlinear system identification based on correlation functions and radial basis functions neural networks / H. V. Ayala, L. D. Coelho // Mechanical Systems and Signal Processing. – 2016. – Vol. 68, Issue 6. – P. 376–378. DOI: 10.1016/j.ymssp.2015.05.022.

27. Abraham A. Swarm intelligence in data mining / A. Abraham, G. Grosan. – Berlin : Springer, 2006. – 267 p. DOI: 10.1007/978-3-540-34956-3.

28. Haroon Shakirat Oluwatosin Client-Server Model / Haroon Shakirat Oluwatosin // IOSR Journal of Computer Engineering. – 2014. – Volume 16, Issue 1. – P. 67–71. DOI: 10.9790/0661-16195771

29. Taylor R.N. Software Design and Architecture. The once and future focus of software engineering / R.N. Taylor, A. van der Hoek // International Conference on Software Engineering, 2007. – P. 226–243. DOI: 10.1109/FOSE.2007.21

30. An object-oriented architecture for extensible structural design software / [Rory Clune, Jerome J. Connor, John A. Ochsendorf, Denis Kelliher] // Computers & Structures. – 2012. – P. 1–17. DOI: 10.1016/j.compstruc.2012.02.002

31. Bernd Bruegge Object-Oriented Software Engineering Using UML, Patterns, and Java (3rd Edition) / Bernd Bruegge, Allen H. Dutoit. – Pearson, 2009. – 800 p. ISBN: 978-0136061250.

32. Qing Li Modeling and Analysis of Enterprise and Information Systems: From Requirements to Realization / Qing Li, Yu-Liu Chen. – Springer Berlin Heidelberg, 2009. – 405 p. DOI: 10.1007/978-3-540-89556-5

33. Hassan Gomaa Designing Software Product Lines with UML 2.0: From Use Cases to Pattern-Based Software Architectures / Hassan Gomaa. – Springer, 2006. – 440 p. DOI: 10.1109/SPLINE.2006.1691600

34. Making database systems usable / [H. V. Jagadish, Adriane Chapman, Aaron Elkiss, et al] // International Conference on Management of Data. – 2007. – P. 13–24. DOI: 10.1145/1247480.1247483

35. Sumathi S. Fundamentals of Relational Database Management Systems / S. Sumathi, S. Esakkirajan. – Springer-Verlag Berlin Heidelberg, 2007. – 792 p. DOI: 10.1007/978-3-540-48399-1

36. Lu Qin Keyword search in databases: the power of RDBMS / Lu Qin, Jeffrey Xu Yu, Lijun Chang // International Conference on Management of Data. – 2009. – P. 681–693. DOI: 10.1145/1559845.1559917

37. Subbotin S. Individual prediction of the hypertensive patient condition based on computational intelligence / S. Subbotin, A. Oliinyk, S. Skrupsky // Information and Digital Technologies : International Conference IDT'2015, Zilina, 7–9 July 2015 : proceedings of the conference. – Zilina : Institute of Electrical and Electronics Engineers, 2015. – P. 336–344. DOI: 10.1109/DT.2015.7222996.

38. Oliinyk A. O. Experimental Investigation with Analyzing the Training Method Complexity of Neuro-Fuzzy Networks Based on Parallel Random Search / A. O. Oliinyk, S. Yu. Skrupsky, S. A. Subbotin // Automatic Control and Computer Sciences. – 2015. – Vol. 49, Issue 1. – P. 11–20. DOI: 10.3103/S0146411615010071.

39. Intelligent information technology of automated diagnostic and pattern recognition systems development: Monograph / [S. A Subbotin, A. Oliinyk, Ye. Gofman et all]. – Kharkov : "Company Smith", 2012. – 317 p. (In russian).

40. Smith S. Genetic and Evolutionary Computation: Medical Applications / S. Smith, S. Cagnoni. – Chicehster : John Wiley & Sons, 2011. – 250 p. DOI: 10.1002/9780470973134.

41. Michael Creel Multi-core CPUs, Clusters, and Grid Computing: A Tutorial / Michael Creel, William L. Goffe // Computational Economics, 2008. – Volume 32, Issue 4. – P. 353–382. DOI: 10.1007/s10614-008-9143-5

42. Kshitij Gupta A study of Persistent Threads style GPU programming for GPGPU workloads / Kshitij Gupta, Jeff A. Stuart, John D. Owens // Parallel Computing. – 2012. – P. 1–14. DOI: 10.1109/InPar.2012.6339596

43. NVIDIA CUDA Compute Unified Device Architecture 5.5. Santa Clara : NVIDIA Corporation, 2014. – 117 p.

Олійник А. О.[1], Субботін С. О.[2], Скрупський С. Ю.[3], Льовкін В. М.[4], Зайко Т. А.[5]

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

[2]Д-р техн. наук, завідувач кафедри програних засобів, Запорізький національний технічний університет, Запоріжжя, Україна

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

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

[5]Канд. техн.наук, ст. викладач кафедри програних засобів, Запорізький національний технічний університет, Запоріжжя, Україна

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ СИНТЕЗУ ДІАГНОСТИЧНИХ МОДЕЛЕЙ НА ОСНОВІ ПАРАЛЕЛЬНИХ ОБЧИСЛЕНЬ

Актуальність. Вирішено задачу автоматизації процесу синтезу діагностичних моделей при обробці великих масивів даних на основі паралельних обчислень. Об'єкт дослідження – процес синтезу діагностичних моделей. Предмет дослідження – методи та інформаційні технології синтезу діагностичних моделей.

Мета роботи полягає в створенні інформаційної технології синтезу діагностичних моделей.

Метод. Запропоновано інформаційну технологію синтезу діагностичних моделей, що представляє собою сукупність діаграм, які описують у графічному вигляді структурні елементи системи, а також поведінкові аспекти їх взаємодії на різних етапах побудови моделей об'єктів діагностування. Запропонована інформаційна технологія дозволяє виконувати побудову розподілених систем діагностування, в яких обчислювально складні етапи синтезу діагностичних моделей виконуються на високопродуктивному серверному обладнанні, що дозволяє істотно підвищити практичний поріг використання систем діагностування при обробці великих масивів даних, здатних вирішувати завдання редукції даних навчальної вибірки, видобування правил, побудови і донавчання діагностичних моделей.

Результати. Розроблено програмне забезпечення, яке реалізує запропоновану інформаційну технологію і дозволяє синтезувати діагностичні моделі на основі заданих наборів даних.

Висновки. Проведені експерименти підтвердили працездатність запропонованої інформаційної технології і дозволяють рекомендувати її для використання на практиці при обробці великих масивів даних для технічного і біомедичного діагностування. Перспективи подальших досліджень можуть полягати в модифікації розробленої технології шляхом впровадження в неї інших методів синтезу діагностичних моделей.

Ключові слова: вибірка даних, діагностування, видобування правил, відбір ознак, паралельні обчислення, синтез моделей.

Олейник А. А.[1], Субботин С. А.[2], Скрупский С. Ю.[3], Левкин В. Н.[4], Зайко Т. А.[5]

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

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

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

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

[5]Канд. техн. наук, ст. преподаватель кафедры программных средств, Запорожский национальный технический университет, Запорожье, Украина

ИНФОРМАЦИОННАЯ ТЕХНОЛОГИЯ СИНТЕЗА ДИАГНОСТИЧЕСКИХ МОДЕЛЕЙ НА ОСНОВЕ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ

Актуальность. Решена задача автоматизации процесса синтеза диагностических моделей при обработке больших массивов данных на основе параллельных вычислений. Объект исследования – процесс синтеза диагностических моделей. Предмет исследования – методы и информационные технологии синтеза диагностических моделей.

Цель работы заключается в создании информационной технологии синтеза диагностических моделей.

Метод. Предложена информационная технология синтеза диагностических моделей, представляющая собой совокупность диаграмм, описывающих в графическом виде структурные элементы системы, а также поведенческие аспекты их взаимодействия на различных этапах построения моделей объектов диагностирования. Предложенная информационная технология позволяет выполнять построение распределенных систем диагностирования, в которых вычислительно сложные этапы синтеза диагностических моделей выполняются на высокопродуктивном серверном оборудовании, что позволяет существенно повысить практический порог использования систем диагностирования при обработке больших массивов данных, способных решать задачи редукции данных обучающей выборки, извлечения правил, построения и дообучения диагностических моделей.

Результаты. Разработано программное обеспечение, которое реализует предложенную информационную технологию и позволяет синтезировать диагностические модели на основе заданных наборов данных.

Выводы. Проведенные эксперименты подтвердили работоспособность предложенной информационной технологии и позволяют рекомендовать ее для использования на практике при обработке больших массивов данных для технического и биомедицинского диагностирования. Перспективы дальнейших исследований могут заключаться в модификации разработанной технологии путем внедрения в нее других методов синтеза диагностических моделей.

Ключевые слова: выборка данных, диагностирование, извлечение правил, отбор признаков, параллельные вычисления, синтез моделей.

REFERENCES

1. Price C. Computer based diagnostic systems. London, Springer, 1999, 136 p. DOI: 10.1007/978-1-4471-0535-0.
2. Bow S. Pattern recognition and image preprocessing. New York, Marcel Dekker Inc., 2002, 698 p. DOI: 10.1201/9780203903896.
3. Sammut C., Webb G. I. eds. Encyclopedia of machine learning. New York, Springer, 2011, 1031 p. DOI: 10.1007/978-0-387-30164-8.
4. 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.
5. 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 ; № 383). DOI: 10.1007/978-0-387-92907-1.
6. 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.
7. Subbotin S., Oliinyk A. The Dimensionality Reduction Methods Based on Computational Intelligence in Problems of Object Classification and Diagnosis, *Recent Advances in Systems, Control*

and Information Technology. Advances in Intelligent Systems and Computing, 2017, Vol. 543, pp. 11–19. DOI: 10.1007/978-3-319-48923-0_2.

8. Subbotin S., Oliinyk A. The Sample and Instance Selection for Data Dimensionality Reduction, *Recent Advances in Systems, Control and Information Technology. Advances in Intelligent Systems and Computing*, 2017, Vol. 543, P. 97–103. DOI: 10.1007/978-3-319-48923-0_13.

9. Oliinyk A. A., Skrupsky S. Yu., Shkarupylo V. V., Blagodariov O. Parallel multiagent method of big data reduction for pattern recognition, *Radio Electronics, Computer Science, Control*, 2017, No. 2, pp. 82–92.

10. Oliinyk A. Production rules extraction based on negative selection, *Radio Electronics, Computer Science, Control*, 2016, Vol. 1, pp. 40–49. DOI: 10.15588/1607-3274-2016-1-5.

11. 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.

12. Oliinyk A., Subbotin S. A. A stochastic approach for association rule extraction, *Pattern Recognition and Image Analysis*, 2016, Vol. 26, No. 2, pp. 419–426. DOI: 10.1134/S1054661816020139.

13. Oliinyk A. O., Skrupsky S. Yu., Subbotin S. A. Using Parallel Random Search to Train Fuzzy Neural Networks, *Automatic Control and Computer Sciences*, 2014, Vol. 48, Issue 6, pp. 313–323. DOI: 10.3103/S0146411614060078.

14. 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.

15. Oliinyk A. A. Skrupsky S. Yu., Shkarupylo V. V., Subbotin S. A. The model for estimation of computer system used resources while extracting production rules based on parallel computations, *Radio Electronics, Computer Science, Control*, 2017, No. 1, pp. 142–152. DOI: 10.15588/1607-3274-2017-1-16.

16. Oliinyk A., Skrupsky S., Subbotin S. Parallel Computer System Resource Planning for Synthesis of Neuro-Fuzzy Networks, *Recent Advances in Systems, Control and Information Technology. Advances in Intelligent Systems and Computing*, 2017, Vol. 543, pp. 88–96. DOI: 10.1007/978-3-319-48923-0_12.

17. David Kirk, Hwu Wen-mei Programming Massively Parallel Processors 3rd Edition. A Hands-on Approach, 2016, 576 p. ISBN: 9780128119860.

18. Andrew Pavlo, Erik Paulson, Alexander Rasin, Daniel J. Abadi, David J. DeWitt A comparison of approaches to large-scale data analysis, *International Conference on Management of Data*, 2009, pp. 165–178. DOI: 10.1145/1559845.1559865

19. Luiz Andre Barroso, Urs Hoelzle The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines, *Synthesis Lectures on Computer Architecture*, 2009, Volume 4, Issue 1, pp. 154. DOI: 10.2200/S00193ED1V01Y200905CAC006

20. Zaigham Mahmood Data Science and Big Data Computing: Frameworks and Methodologies, *Springer International Publishing*, 2016, P. 332. DOI: 10.1007/978-3-319-31861-5

21. Salfner F., Lenk M., Malek M. A survey of online failure prediction methods, *ACM computing*, 2010, Vol. 42, Issue 3, pp. 1–42. DOI: 10.1145/1670679.1670680.

22. Shin Y. C., Xu C. Intelligent systems : modeling, optimization, and control. Boca Raton, CRC Press, 2009, 456 p. DOI: 10.1201/9781420051773.

23. Bishop C. M. Pattern recognition and machine learning. New York, Springer, 2006, 738 p.

24. Gen M., Cheng R. Genetic algorithms and engineering design. New Jersey, John Wiley & Sons, 1997, 352 p. DOI: 10.1002/9780470172254.

25. Yu X., Gen M. Introduction to Evolutionary Algorithms (Decision Engineering). London, Springer, 2010, 418 p. DOI: 10.1007/978-1-84996-129-5.

26. Ayala H. V., Coelho L. D. Cascaded evolutionary algorithm for nonlinear system identification based on correlation functions and radial basis functions neural networks, *Mechanical Systems and Signal Processing*, 2016, Vol. 68, Issue 6, pp. 376–378. DOI: 10.1016/j.ymssp.2015.05.022.

27. Abraham A., Grosan G. Swarm intelligence in data mining. Berlin, Springer, 2006, 267 p. DOI: 10.1007/978-3-540-34956-3.

28. Haroon Shakirat Oluwatosin Client-Server Model, *IOSR Journal of Computer Engineering*, 2014, Volume 16, Issue 1, pp. 67–71. DOI: 10.9790/0661-16195771

29. Taylor R. N., A. van der Hoek Software Design and Architecture. The once and future focus of software engineering, *International Conference on Software Engineering*, 2007, pp. 226–243. DOI: 10.1109/FOSE.2007.21

30. Rory Clune, Jerome J. Connor, John A. Ochsendorf, Denis Kelliher An object-oriented architecture for extensible structural design software, *Computers & Structures*, 2012, pp. 1–17. DOI: 10.1016/j.compstruc.2012.02.002

31. Bernd Bruegge, Allen H. Dutoit. Object-Oriented Software Engineering Using UML, Patterns, and Java (3rd Edition). Pearson, 2009, 800 p. ISBN: 978-0136061250.

32. Qing Li, Chen Yu-Liu Modeling and Analysis of Enterprise and Information Systems: From Requirements to Realization. Springer Berlin Heidelberg, 2009, 405 p. DOI: 10.1007/978-3-540-89556-5

33. Hassan Gomaa Designing Software Product Lines with UML 2.0: From Use Cases to Pattern-Based Software Architectures. Springer, 2006, 440 p. DOI: DOI: 10.1109/SPLINE.2006.1691600

34. Jagadish H. V., Adriane Chapman, Aaron Elkiss, Magesh Jayapandian, Yunyao Li Making database systems usable, *International Conference on Management of Data*, 2007, pp. 13–24. DOI: 10.1145/1247480.1247483

35. Sumathi S., Esakkirajan S. Fundamentals of Relational Database Management Systems. Springer-Verlag Berlin Heidelberg, 2007, 792 p. DOI: 10.1007/978-3-540-48399-1

36. Lu Qin, Jeffrey Xu Yu, Lijun Chang Keyword search in databases: the power of RDBMS, *International Conference on Management of Data*, 2009, pp. 681–693. DOI: 10.1145/1559845.1559917

37. Subbotin S., Oliinyk A., Skrupsky S. Individual prediction of the hypertensive patient condition based on computational intelligence, *Information and Digital Technologies : International Conference IDT'2015, Zilina, 7–9 July 2015 : proceedings of the conference*. Zilina, Institute of Electrical and Electronics Engineers, 2015, pp. 336–344. DOI: 10.1109/DT.2015.7222996.

38. Oliinyk A. O., Skrupsky S. Yu., Subbotin S. A. Experimental Investigation with Analyzing the Training Method Complexity of Neuro-Fuzzy Networks Based on Parallel Random Search, *Automatic Control and Computer Sciences*, 2015, Vol. 49, Issue 1, pp. 11–20. DOI: 10.3103/S0146411615010071.

39. 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).

40. Smith S., Cagnoni S. Genetic and Evolutionary Computation: Medical Applications. Chicehster, John Wiley & Sons, 2011, 250 p. DOI: 10.1002/9780470973134.

41. Michael Creel, William L. Goffe Multi-core CPUs, Clusters, and Grid Computing: A Tutorial, *Computational Economics*, 2008, Volume 32, Issue 4, pp. 353–382. DOI: 10.1007/s10614-008-9143-5

42. Kshitij Gupta, Jeff A. Stuart, John D. Owens A study of Persistent Threads style GPU programming for GPGPU workloads, *Parallel Computing*, 2012, pp. 1–14. DOI: 10.1109/InPar.2012.6339596

43. NVIDIA CUDA Compute Unified Device Architecture 5.5. Santa Clara, NVIDIA Corporation, 2014, 117 p.