ISSN 2278-3091
Volume 8, No.5, September - October 2019
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse77852019.pdf
https://doi.org/10.30534/ijatcse/2019/77852019

Computational Complexity of the Accessory Function Setting Mechanism in Fuzzy Intellectual Systems

M. Ayaz Ahmad[1], Irina Tvoroshenko[2], Jalal Hasan Baker[3], Vyacheslav Lyashenko[4]
[1]Department of Physics, Faculty of Science, University of Tabuk, Saudi Arabia, mayaz.alig@gmail.com
[2] Department of Informatics, Kharkiv National University of RadioElectronics, Ukraine, iryna.tvoroshenko@nure.ua
[3] Department of Physics, Faculty of Science, University of Tabuk, Saudi Arabia, jhasan@ut.edu.sa
[4] Department of Informatics, Kharkiv National University of RadioElectronics, Ukraine, lyashenko.vyacheslav@gmail.com

ABSTRACT

This article proposes a mechanism for processing fuzzy symptoms, as well as all related factors presented in natural language, which made it possible to improve the intellectual decision-making system in the field of cardiology by adjusting membership functions. The computational complexity of solving the problem according to the criterion of time costs is determined, it is close to exponential from the selected calculation accuracy, which indicates the effectiveness and the absence of disadvantages of the proposed mechanism for a given criterion. The software implementation of the mechanism for setting membership functions in a fuzzy intellectual system is implemented in the Python 3.6 object-oriented programming environment. The laboratory operation of a fuzzy intellectual system has confirmed the high reliability of making objective decisions in the medical subject area.

Key words: Intelligent System, Cardiology, Linguistic Variables, Term, Fuzzy Logic, Membership Function, Gradient Method, Genetic Algorithm, Dichotomy Method, Computational Complexity, Criterion.

1. INTRODUCTION

Intelligent decision-making system is based on knowledge, with the help of which it makes assumptions about possible alternative solutions [1]–[5]. The logical structure of this system provides the issuance of recommendations to clarify the decision [6], and also makes decisions whose level of competence is not lower than the level of a human expert [7]. The objectives of the intelligent system are:

– adoption of a set of formal and heuristic knowledge from high-level specialists;

– the use of applied knowledge by other specialists in this subject area or related professions;

– providing advice aimed at improving the level of decisions [8].

As a rule, intelligent systems operate using these components [6]:

– knowledge base, which includes the knowledge, experience and intuition of the expert;

– database;

– block logical inference;

– a block of explanations;

– friendly interface.

The complex system presented in [1], [6–8] operates under conditions of uncertainty and severe resource constraints [9], and therefore includes additional modules in its structure [6].

To assess the quality of the developed models and methods for modeling complex processes [8], to select a mechanism for setting membership functions and determining its computational complexity, a practical implementation of information technologies in the medical subject field, namely cardiology, has been proposed [10].

Cardiology studies cardiovascular diseases and is an extremely important area in the development of modern science [11], as it studies diseases that have become the main cause of disability and premature death in residents of economically developed countries. The share of such diseases in the mortality structure is from 40% to 60%.

There is a constant increase in indicators in the number of diseases, doctors record the young age of patients with problems of the cardiovascular system, but thanks to the constant development of information technology, it became possible to completely cope with heart diseases that were considered incurable yesterday [12].

Definition of the disease is a complex creative process. It requires a large amount of knowledge from the doctor, the ability to think clinically, inexhaustible interest and attention to the patient. Only professionals with extensive experience in a field such as cardiology can timely make the correct diagnosis.

The most important priority for the doctor always remains the sequence of development of the signs of the disease, established by questioning the patient and with dynamic medical supervision [13]. In addition, changes detected during laboratory and instrumental examination usually have different causes. A gross deviation of any indicator from the norm is of great screening importance for establishing the disease.

This article makes a choice of a mechanism for setting membership functions in fuzzy intelligent systems and determining its computational complexity.

2. RESEARCH PROBLEM STATEMENT

Given the fuzzy nature of cardiological symptoms, as well as all related factors, the description of the subject area was made in natural language by certain terms of fuzzy logic [8]. For example, for the linguistic variable “temperature”, the terms are “high”, “normal”, and “lowered.”

The implementation of linguistic variables provides physical descriptions of their terms, the definition of which is possible when working together with expert doctors. From 3 to 7 terms per variable is sufficient for an accurate representation of the physical quantity.

The belonging of each exact value to one of the terms of a linguistic variable is determined by the membership function [14].

As a rule, standard membership functions are used [15]. When solving specific problems, one can choose other, more suitable types of membership functions and, as practice shows, achieve better results of the intellectual system than using standard-type functions.

Suppose that some membership function is given. An expert doctor sets her appearance $\mu(x)$ with some parameters k. In the process of fuzzy inference, for example, Zade-Mamdani [16] it is determined that the original membership function has an error Δ, which exceeds some margin of error ε.

It is necessary:
– on many functions $\mu(x)$ offer a mechanism for setting membership functions that minimizes Δ subject to margin of error $\varepsilon \le \varepsilon^*$, where ε^* – error limit;
– to determine and justify the computational complexity of solving the problem by the time criterion τ.

3. MECHANISMS FOR CONFIGURING VARIOUS KINDS OF MEMBERSHIP FUNCTIONS

The mechanism for setting membership functions in fuzzy intelligent systems should take into account the presence of:
– a special apparatus for storing and processing knowledge represented by symbolic information;
– the ability to solve the intellectual non-computational nature of the problem, requiring access to data semantics;
– the ability to explain and interpret the result of a complex system with a certain confidence coefficient.

To find the value of the confidence coefficient, it is necessary to construct a certain membership function. There is a problem of searching for parameters of a given function and bringing them into line with existing expert estimates.

Consider three ways to search for membership function parameters:
– gradient methods;
– genetic algorithms;
– binary search method.

Gradient methods suggest that any set of real numbers $(v_1, v_2, ..., v_k)$, taken in a certain order can be considered as a point or vector with the same coordinates in k-dimensional space [17].

View record $v = (v_1, v_2, ..., v_k)$ indicates a point or vector v with coordinates indicated in brackets.

If for k-dimensional vectors and basic algebraic operations:
– addition and subtraction

$$v \pm w = (v_1 \pm w_1, v_2 \pm w_2, ..., v_\kappa \pm w_k);$$

– multiplication by a real number u

$$u \times v = (u \times v_1, u \times v_2, ..., u \times v_k);$$

– scalar product

$$v \times w = (v_1 \times w_1, v_2 \times w_2, ..., v_\kappa \times w_k),$$

then the set of such vectors denote E_k and called k-dimensional Euclidean space.

Vector length v called the number determined by the formula:

$$|v| = \sqrt{vv} = \sqrt{v_1^2 + v_2^2 + ... + v_k^2}. \quad (1)$$

It is important that the length of the vector (1) can be calculated only when its components are presented in the same measurement scale or are dimensionless quantities.

Also, if the product $v \times w = 0$ at $|v| \neq 0$ and $|w| \neq 0$, then vectors v and w are orthogonal.

The unit vector is determined by the formula:

$$t = (t_1, t_2, ..., t_k) = \left(\frac{v_1}{|v|}, \frac{v_2}{|v|}, ..., \frac{v_k}{|v|} \right). \qquad (2)$$

Let in E_k given some point $v = (v_1, v_2, ..., v_k)$, unit vector t and a function continuously differentiable with respect to all arguments $f(V) = (v_1, v_2, ..., v_k)$.

Derivative at point V from function $f(V)$ in the direction of the beam defined by the vector t, called the limit:

$$\frac{\partial f(V)}{\partial t} = \lim \frac{f(v_1 + \lambda t_1, ..., v_k + \lambda t_k) - f(v_1, ..., v_k)}{\lambda} \Big| \lambda \to 0$$

or

$$\frac{\partial f(V)}{\partial t} = \left[\frac{\partial f(V)}{\partial v_1} t_1, \frac{\partial f(V)}{\partial v_2} t_2, ..., \frac{\partial f(V)}{\partial v_k} t_k \right]. \qquad (3)$$

Gradient function $f(V)$ called vector $\Delta f(V)$ with coordinates equal to the partial derivative with respect to the corresponding arguments:

$$\nabla f(V) = \left(\frac{\partial f(V)}{\partial v_1}, \frac{\partial f(V)}{\partial v_2}, ..., \frac{\partial f(V)}{\partial v_k} \right). \qquad (4)$$

The gradient indicates the direction of greatest increase in function.

Opposite direction $-\Delta f(V)$ called anti-gradient, it shows the direction of the steepest decrease in function.

At the point of extremum V^* gradient is zero $\Delta f(V^*) = 0$. If it is impossible to determine analytically, then they are calculated approximately:

$$\frac{\nabla f(V)}{\nabla v_i} = \frac{\Delta f(V)}{\Delta v_i}, \qquad (5)$$

where $\Delta f(V)$ – function increment $f(V)$ when changing the argument by Δv_i. Moving along the gradient (anti-gradient), you can achieve the maximum (minimum) of the function. This is the essence of the gradient optimization method.

One of the main problems of applying the gradient search method (1–5) is the choice of the value of each discrete step. Steps can be constant or variable. The second option in the implementation of the algorithm is more complex, but usually requires less iteration.

The considered method is used only for nonlinear functions. If the function is linear, then the choice of the optimal parameter value is difficult.

The disadvantage of these methods is their computational resource consumption and the difficulty of finding a global extremum in the presence of a function with many local extremes.

A genetic algorithm is a heuristic search method used to solve optimization and modeling problems. It is carried out by sequential selection, combination and variation of the desired parameters using mechanisms reminiscent of biological evolution. The method is a type of evolutionary computation. A distinctive feature of genetic algorithms is the emphasis on the use of the “crossing” operator, which performs the operation of recombination of candidate solutions, the role of which is similar to the role of crossing in wildlife [18].

In a genetic algorithm, a task is encoded so that its solution can be represented as a vector. A number of initial vectors are randomly generated. They are estimated using the “fitness function” and a specific value is assigned to each vector, which determines the probability of survival of the organism represented by this vector.

Using fitness values, vectors that are allowed to “cross” are selected. “Genetic operators” are applied to these vectors, thus creating the next “generation”. Individuals of the next generation are also evaluated, then selection is made, genetic operators are used, etc. Thus, an “evolutionary process” with several life cycles is simulated until the criterion for stopping the algorithm is met.

Genetic algorithms search for solutions in very large and complex state spaces. They allow us to approximate the function with fairly high accuracy, although they also have a number of significant drawbacks:
– high computational complexity (which is irrational in machine processing);
– difficulty in implementing the algorithm itself;
– instability of decisions.

The binary search method implements the dichotomy method (assumes half division), which can be used to solve the problem of selecting the parameters of fuzzy membership functions. A dichotomy is the sequential division of a certain set into two parts that are not interconnected. Dichotomous division is a way of forming mutually exclusive classes of one concept or term, which is necessary to create a classification of selected elements [19].

One of the advantages of dichotomous division is the simplicity of its implementation. Two classes are enough that exhaust the scope of a divisible concept.

Thus, dichotomous division is always proportional, the members of the division are mutually exclusive. Each object of a divisible set falls into only one of the classes a or no a. The division is carried out on one basis – the presence or absence of some attribute. Denoting a divisible concept a and highlighting in its volume some kind b, can divide volume a in two parts b and not b.

The dichotomy method is somewhat similar to the binary search method, but differs from it by the criterion of dropping ends [19].

Let a membership function of the form

$$f(x):[a,b]\to R, f(x)\in C([a,b]). \tag{6}$$

We divide the given segment in half and take two points symmetric with respect to the center x_1 and x_2 so that:

$$x_1=\frac{a+b}{2}-\delta; \quad x_2=\frac{a+b}{2}+\delta, \tag{7}$$

where δ – some number in the interval $\left(0,\frac{b-a}{2}\right)$.

We discard one of the ends of the initial interval to which one of the two newly set points with the maximum value turned out to be closer (minimum search). If $f(x_1)>f(x_2)$, then a segment is taken $[x_1,b]$, and the segment $[a,x_1]$ discarded. Otherwise, a segment is taken that is mirrored relative to the middle $[a,x_2]$, but discarded $[x_2,b]$.

The ends drop procedure, the computational dichotomy procedures, and the fuzzy inference procedures on some of the functions (6) are performed until the specified accuracy is achieved ε on the set of values $\{k_i\}$, $i\in I$:

$$\left|y_{actual}-y_{expected}\right|\le\varepsilon, \tag{8}$$

where ε – the numerical value of the approximation accuracy, that is, the maximum permissible deviation of the actual value from the expected.

Since it is necessary to calculate new points at each iteration, it is necessary to propose a mechanism that will allow calculating only one new point at the next iteration, which significantly optimizes the procedure. This approach is achieved by mirror division of the segment in the golden section (golden section method) using the parameter

$$\delta=(b-a)\left(\frac{1}{2}-\frac{1}{\phi}\right). \tag{9}$$

It should be noted that a dichotomy has the property of convergence for any continuous functions, including non-differentiable ones [19].

The mechanism for selecting the parameters of the fuzzy membership function based on the dichotomy method is as follows:
– when defuzzification fuzzy membership functions, some values are included in the product “If X, then Y”;
– the argument of the input membership function is a certain quantity x_{input}.

According to the defuzzification method, this value is projected onto the output function Y. As an output, we get some value y_{actual};
– the essence of the selection of parameters is to change the appearance of the output membership function in accordance with the expected value received from the expert $y_{expected}$.

Since the membership function of any type has a certain parametric value k, that is, it is subject to selection of value. For example, for a function of “large” magnitude, the expression $y=1-e^{k(x-a)^2}$.

Parameter k undergoes a dichotomy using an approximation step equal to $\frac{\varepsilon}{2}$. This step value is due to the need to execute the expression (8).

At the output, this mechanism provides the found values of the membership function parameters.

4. COMPUTATIONAL COMPLEXITY OF MEMBERSHIP FUNCTION SETTINGS

The study of the computational complexity of the mechanism for providing functional capabilities showed that the number of iterative parameters of the dichotomy using the golden section method, respectively, and its operating time τ have an exponential dependence on the value of the accuracy value ε.

Indeed, a search in one iteration contains costs $\approx n$ operations. A dichotomy strategy involves dividing into two when implementing some iteration.

Division of integer nonzero n – bit (not including signed digits) numbers A and B, presented in the direct (for simplicity) code leads to an integer quotient C and the whole remainder 0, which is assigned the sign of the dividend, and the sign of the quotient is calculated as the sum modulo 2 operands A and B.

The division is performed in the following sequence [20]:

a) divider B shifts to the left (normalizes) so that there is 1 in the senior information category, then the number of shifts is calculated S. The quotient from division can be no more than $S+1$ non-zero digits;

b) performed $S+1$ module division cycle $|A|$ on $|B'|$, where $|B'|$ – normalized B. The result of this operation is $(S+1)$ rank private starting from the oldest of $(S+1)$ junior;

c) residue obtained in the last division cycle R_{S+1} moves right by S discharges if it is positive. In case of receiving a negative balance $R_{S+1} < 0$ is added to it for recovery $|B'|$, i.e:

$$[R_{S+1}]_{vost} = R_{S+1} + |B'|. \tag{10}$$

After that, a shift to the right by S discharges. The result is the remainder of the division.

To determine the upper bound for the complexity of the dichotomy method for operations at each iteration, we present the search strategy in the form:

$$n + n(n-1) + n(n-1)(n-2) + n(n-1)(n-2)(n-3) + \ldots \tag{11}$$

We take into account the expansion in a series of exponential functions [21] as

$$e^x \approx 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \ldots \quad . \tag{12}$$

Then, in the field of fine-tuning the parameters of membership functions in fuzzy logic inference problems [15], one can determine an upper bound for complexity that does not exceed the following dependence:

$$O(n) \approx ke^n, \tag{13}$$

where $O(n)$ – method complexity; n – number of computational iterations; k – a parameter determined by the computing resources of the computer, in this case k some nonlinear increasing function.

Obviously, dependence (13) is largely correlated with the parameter k, which requires, perhaps, solving optimization problems for choosing the computer hardware configuration.

Then the solution to the problem can be represented as:

$$\begin{aligned} &\varepsilon \to \min; \\ &V \geq V^*; \\ &\varepsilon \leq \varepsilon^*, \end{aligned} \tag{14}$$

where V^* – processor speed limit set in advance.

The solution to problem (14) is nontrivial, since the dependencies ε and V – nonlinear. Given that in (13)

$$n \approx \frac{1}{\varepsilon},$$

then solution (14) simultaneously reduces complexity (13), which is important in practical applications.

To reduce complexity, the use of the branch and bound method [22] can be proposed, which is used to find optimal solutions to various optimization problems, especially discrete and combinatorial optimization. In essence, the method is a variation of exhaustive search with elimination of subsets of feasible solutions that obviously do not contain optimal solutions. This leads to a significant decrease in complexity (13). The method ends when it reaches $\varepsilon \leq \varepsilon^*$.

For a graphical assessment of computational complexity, the accuracy values and the number of operations were normalized to a unit norm. If you designate accuracy ε', then it takes values $\varepsilon' = 1 - \varepsilon$.

The number of operations is calculated by multiplying the real number of executable operators by the proportionality coefficient a, where $a = \frac{1}{N}$, N – number of operations $\varepsilon' \geq 1$.

At accuracy value $\varepsilon = 0.03$, application computing time was: $\tau = 14$ ms ; at accuracy value $\varepsilon = 0.002$, application computing time was: $\tau = 886$ ms .

5. SOFTWARE IMPLEMENTATION OF THE MECHANISM FOR SETTING MEMBERSHIP FUNCTIONS IN FUZZY INTELLIGENT SYSTEMS

The application is designed to implement automatic configuration of membership function parameters k according to some input conditions.

The user has the ability to determine:
– types of input and output fuzzy-quantities ("small", "medium", "large");
– argument value for input quantity x_{input} (defuzzification benchmark);
– expected value of output argument $y_{expected}$ based on some previous expert opinions;
– required accuracy ε.

The essence of the calculations is the selection of the coefficient k, determining the type of function of the output fuzzy quantity of a given type in order to find an approximation of the value y_{actual} to value $y_{expected}$ in compliance with the requirements of accuracy of calculations.

The application implements an algorithm for setting parameters for defuzzification of fuzzy values:
– determination of the types of input and output fuzzy values;
– indication of the reference point as an argument of the function of the input fuzzy value x_{input};
– finding the value of the function of the input fuzzy value corresponding to a given reference point and matching the value of the output fuzzy value;
– formation of a composite integrand figure by a limited function of the output fuzzy value and a line of projecting the value of the function of the input fuzzy value on the graph of the function of the output fuzzy value;
– finding the abscissa of the center of gravity of the resulting figure, which determines y_{actual}.

The mechanism for selecting the coefficient value for the output fuzzy quantity is reduced to minimizing (with a previously determined accuracy) the absolute difference between the expected and actual value of the argument: $\Delta x = \left| y_{expected} - y_{actual} \right|$.

The software implementation of this mechanism involves some approximating assumptions that allow one to implement calculations by numerical methods [23] by dividing the analyzed integrand into a number of simple components to find the center of gravity.

The class hierarchy of a software implementation consists of three base classes:
– TMainForm interface form class;
– FuzzyValue fuzzy class;
– class of an arbitrary polygon MyPoly, used to find the center of gravity in the process of defuzzification.

Aggregation relation is established between all classes of the program. The computational component is defined in the class functions.

The functions for implementing the proposed mechanism are the function of finding the center of gravity getMidPointX (…) and the function of dichotomous selection of the selectionOfK(…) parameter.

The developed program implements the calculation of the complexity and operating time of the intelligent system, depending on the selected accuracy. The opportunity to save the results of work and some other service functions.

6. CONCLUSION

The presented mechanism for processing fuzzy cardiological symptoms, as well as all related factors, allows cardiology workers, despite their length of service and work experience, to make a professional diagnosis with a high degree of reliability. In advance to warn the patient about the danger and identify factors that contribute to the elimination of the disease process.

The proposed mechanism has improved the intellectual decision-making system in the field of cardiology by adjusting membership functions.

The binary search mechanism is automated, which involves the joint work of the dichotomy and golden section methods, as well as setting the required accuracy of the selection of parameters and speed.

The computational complexity of solving the problem by the criterion of time costs is determined. It is close to exponential from the selected calculation accuracy, which indicates the effectiveness and absence of disadvantages of the proposed mechanism according to a given criterion.

The upper limit of the computational complexity of decision-making processes, which is close to quadratic, is substantiated.

The software implementation of the system was implemented in an object-oriented programming environment Python 3.6 [24].

The pilot operation of hardware and software of a fuzzy intellectual system confirmed the high reliability of making objective decisions on the topic of research.

REFERENCES

1. Ye. I. Kucherenko, V. A. Filatov, I. S. Tvoroshenko, and R. N. Baidan. **Intellectual technologies in decision-making technological complexes based on fuzzy interval logic**, *East European Journal of Advanced Technologies*, Vol. 2, pp. 92–96, 2005.
2. A. Coskun-Setirek, and Z. Tanrikulu. **Intelligent interactive voice response systems and customer satisfaction,** *International Journal of Advanced Trends in Computer Science and Engineering*, Vol. 8, no. 1, pp. 4–11, 2019. https://doi.org/10.30534/ijatcse/2019/02812019
3. P. Orobinskyi, D. Petrenko, and V. Lyashenko. **Novel Approach to Computer-Aided Detection of Lung Nodules of Difficult Location with Use of Multifactorial Models and Deep Neural Networks**, in *15th International Conference on the Experience of Designing and Application of CAD Systems (CADSM)*, 2019, pp. 1-5. https://doi.org/10.1109/CADSM.2019.8779340
4. V. Lyashenko, O. Kobylin, and M. Minenko. **Tools for Investigating the Phishing Attacks Dynamics,** in *International Scientific-Practical Conference Problems of Infocommunications. Science and Technology (PIC S&T)*, 2018, pp. 43-46. https://doi.org/10.1109/INFOCOMMST.2018.8632100
5. D. Fontes, P. A. Pereira, and F. Fontes. **A Decision Support System for TV self-promotion Scheduling**, *International Journal of Advanced Trends in Computer Science and Engineering*, Vol. 8, no. 2, pp. 134-140, 2019. https://doi.org/10.30534/ijatcse/2019/06822019
6. I. S. Tvoroshenko. **Structure and functions of intelligent decision-making tools in complex systems**, *Artificial Intelligence*, Vol. 4, pp. 462–470, 2004.
7. Ye. I. Kucherenko, and I. S. Tvoroshenko. **Operative evaluation of the space of states of complex distributed objects using fuzzy interval logic**, *Artificial Intelligence*, Vol. 3, pp. 382–387, 2011.
8. I. S. Tvoroshenko. **Analysis of Decision-Making Processes**. *Intelligent Systems, Information Processing Systems*, Vol. 2, pp. 248–253, 2010.
9. A. S. Egorov, and A. N. Shaykin. **Logical modeling under uncertainty based on fuzzy interval Petri nets**, *News of the Russian Academy of Sciences, Theory and Control Systems*, Vol. 2, pp. 134–139, 2002.
10. O. G. Avrunin, Ye. V. Bodiansky, M. V. Kalashnik, V. V. Semenets, V. O. and Filatov. **Modern Intelligent Technologies of Functional Medical Diagnostics: monograph**, Kharkiv, KNURE, 2018, pp. 37–55.
11. K. W. Johnson, et al. **Artificial intelligence in cardiology**, *Journal of the American College of Cardiology*, Vol. 71, no. 23, pp. 2668–2679, 2018. https://doi.org/10.1016/j.jacc.2018.03.521
12. R. M. Ratwani, J. Reider, and H. Singh. **A decade of health information technology usability challenges and the path forward**, *Jama*, Vol. 321, no. 8, pp. 743–744, 2019. https://doi.org/10.1001/jama.2019.0161
13. A. Nair, and D. Dreyfus. **Technology alignment in the presence of regulatory changes: The case of meaningful use of information technology in healthcare**, *International journal of medical informatics*, Vol. 110, pp. 42–51, 2018. https://doi.org/10.1016/j.ijmedinf.2017.11.012
14. H. C. Liu, X. Luan, Z. Li, and J. Wu. **Linguistic Petri Nets Based on Cloud Model Theory for Knowledge Representation and Reasoning**, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 30, pp. 717–728, 2018. https://doi.org/10.1109/TKDE.2017.2778256
15. J. H. Zhang, J. J. Xia, J. M. Garibaldi, P. P. Groumpos, and R. B. Wang. **Modeling and control of operator functional state in a unified framework of fuzzy inference Petri nets**, *Computer Methods and Programs in Biomedicine*, Vol. 144, pp. 147–163, 2017. https://doi.org/10.1016/j.cmpb.2017.03.016
16. D. T. Mukhamedieva, and O. M. Begimov. **Methods and Algorithms of Fuzzy Models Construction Assessing the State of the Low-Formalized Processes**, *International Journal of Applied Engineering Research*, Vol. 13, no. 6, pp. 4364–4372, 2018.
17. I. V. Konnov. **Conditional gradient method without line-search**, *Russian Mathematics*, Vol. 62, no. 1, pp. 82–85, 2018. https://doi.org/10.3103/S1066369X18010127
18. A. H. Hamamoto, et al. **Network anomaly detection system using genetic algorithm and fuzzy logic**, *Expert Systems with Applications*, Vol. 92, pp. 390–402, 2018. https://doi.org/10.1016/j.eswa.2017.09.013
19. L. Pickup, A. Lang, S. Atkinson, and S. Sharples. **The dichotomy of the application of a systems approach in UK healthcare the challenges and priorities for implementation**, *Ergonomics*, Vol. 61, no. 1, pp. 15–25, 2018. https://doi.org/10.1080/00140139.2017.1306632
20. A. Oliinyk, S. Skrupsky, and S. Subbotin. **Experimental Research and Analysis of Complexity of Parallel Method for Production Rules Extraction**, *Automatic Control and Computer Sciences*, Vol. 52, no. 2, pp. 89–99, 2018. https://doi.org/10.3103/S0146411618020062

21. X. García-Martínez, and T. Van der Linden. **A characterisation of Lie algebras via algebraic exponentiation**, *Advances in Mathematics*, Vol. 341, pp. 92–117, 2019.
https://doi.org/10.1016/j.aim.2018.10.034
22. B.J. Kubica. **Branch-and-Bound-Type Methods. In Interval Methods for Solving Nonlinear Constraint Satisfaction**, *Optimization and Similar Problems*, pp. 272–45, 2019.
https://doi.org/10.1007/978-3-030-13795-3_4
23. A. Verruijt. **Numerical Solution.** *In An Introduction to Soil Mechanics*, Springer, Cham, 2018, pp. 1412–147.
https://doi.org/10.1007/978-3-319-61185-3
24. K. R. Srinath. **Python – The Fastest Growing Programming Language**, *International Research Journal of Engineering and Technology*, Vol. 4, pp. 354–357.