Method for Calculating the Residual Resource of Fog Node Elements of Distributed Information Systems of Critical Infrastructure Facilities

Anton Zahynei[1], Yurii Shcheblanin[2], Oleg Kurchenko[2], Andrii Anosov[3], and Vladyslav Kruglyk[4]

[1] *State University of Information and Communication Technologies, 7 Solomyanska str*., Kyiv, 03110, Ukraine
[2] *Taras Shevchenko National University of Kyiv, 60 Volodymyrska str., Kyiv, 01033, Ukraine*
[3] *Borys Grinchenko Kyiv Metropolitan University*, *18/2 Bulvarno-Kudriavska str.*, *Kyiv*, *04053*, *Ukraine*
[4] *Bogdan Khmelnitsky Melitopol state pedagogical university*, *59 Naukovogo mistechka*, *Zaporizhzhya*, *69000 Ukraine*

Abstract

The paper considers the process of predicting the residual technical resource of elements (devices) of information and communication systems, which are components of fog computing. The analysis of the influence of the technical state of the elements of fog nodes on the functioning of distributed information systems is carried out. A method for calculating the residual resource of fog nodes has been developed, which is advisable to use for planning the modernization of fog nodes and in the process of calculating the fault tolerance of the entire distributed infrastructure.

Keywords

Fog computing, information systems, model, residual resource.

1. Introduction

Modern technology development and implementation of distributed information systems in critical areas of human activity necessitate the improvement of methods for forecasting and optimizing the use of technical resources. One of the key innovative technologies that has become an important component in modern information and communication systems is fog computing technology [1].

The fog node is the core component of the fog computing architecture. Fog nodes are either physical components (e.g. gateways, switches, routers, servers, etc.) or virtual components (e.g. virtualized switches, virtual machines, cloudlets, etc.) that are tightly coupled with the smart end-devices or access networks, and provide computing resources to these devices [2].

In distributed information systems of critical infrastructure, such as energy, transportation, electronic communications, and others, where security and uninterrupted operation are a priority, the use of fog nodes becomes an important strategic solution [3]. Node elements allow the distributed system to be more flexible and adaptable to the challenges it faces.

Fog nodes consist of devices that cannot process data on their own. Examples of such devices include temperature sensors, humidity sensors, motion sensors, smoke detectors, and video cameras without image processing capabilities.

Given the importance and unique capabilities of fog nodes in the context of critical infrastructure, there is a need to develop a method for calculating the remaining life of node elements [4–6]. This method will allow us to effectively determine and predict the resources remaining after devices have

CPITS-2024: Cybersecurity Providing in Information and Telecommunication Systems, February 28, 2024, Kyiv, Ukraine
EMAIL Antonio.com237@gmail.com (A. Zahynei); sheblanin@ukr.net (Y. Shcheblanin); kurol@ukr.net (O. Kurchenko); a.anosov@kubg.edu.ua (A. Anosov); krugvs@mdpu.org.ua (V. Kruglyk)
ORCID: 0000-0002-0303-8501 (A. Zahynei); 0000-0002-3231-6750 (Y. Shcheblanin); 0000-0001-7671-8287 (O. Kurchenko); 0000-0002-2973-6033 (A. Anosov); 0000-0002-5196-7241 (V. Kruglyk)

performed computing and networking tasks over a long period.

The research aims to develop and optimize fog computing technology for critical infrastructure objects. It will consider the residual resources of node elements to create more sustainable and efficient distributed information systems. The research will also consider failure rate criteria [7].

2. Review of Existing Methods

Most of the research related to the resource of fog node elements is to calculate their required amount to perform the tasks. Therefore, in [8] the authors present a review of resource reservation methods in a fog environment, i.e. the calculation of the amount of resources that will be sufficient for the stable functioning of the information system is investigated. In addition, the paper describes the methods that should be used to calculate the time to failure of fog node elements, such methods include a task execution model, statistical method, etc. [8]. The task execution model is used to calculate the execution time of a specific task by a system element. The model takes into account such factors as the type of task, the size of the input data, and the available resources of the tool [8]. In turn, the statistical method uses statistical data to calculate the execution time of the task. The method takes into account such criteria as the type of task, the size of the task, and the past load on the fog node.

The authors of the study show that the task execution model can be more accurate than the statistical method. This is because the task execution model considers the specific features of the task that can affect its execution time. These features may include the complexity of the task, the skill level of the person performing the task, and the availability of resources.

At the same time, the task execution model can be more difficult to implement than the statistical method. This is due to the fact that the development of a task execution model requires information about the purpose of the distributed system node and its components [9].

A method for calculating the residual resource of fog nodes that takes into account Quality of Service (QoS) requirements is proposed in [10]. The method uses a task execution model to calculate the task execution time, and then uses this execution time to calculate the residual resource.

Implementation of the method consists of the following steps [10]:

QoS Requirements Assessment:

The QoS requirements for the task must first be assessed. QoS requirements may include the following metrics:

- Maximum delay time.
- Maximum error rate.

After calculating the specified metrics, the task execution time is calculated using the task execution model. The task execution model takes into account such criteria as the task type, the task size, and the available resources [11].

Once the task execution time has been calculated, the remaining node element resource can be calculated. The residual resource of a fog node can be calculated by subtracting the resource consumed by the node to perform a task from the total resource of the fog node.

The total node resource requirement for a task is calculated as the product of the node's task execution time and the average resource consumption for one task

The average resource utilization for task execution can be calculated using a statistical method [12].

However, this method does not account for unforeseen events, such as sudden spikes in load or equipment failures. This can lead to fog node elements being unable to provide the necessary technical specifications for task execution. Additionally, the method does not consider the dynamics of node load, which can change over time. This can lead to the fog node being unable to provide the required quality of task execution if the load and number of tasks change [13].

3. Method Development

To solve the problem of forecasting and calculating the remaining life of elements of fog nodes in distributed information systems of critical infrastructure facilities, it is proposed to use probabilistic forecasting under conditions of resource constraints for regular testing of node elements to assess their technical condition. To do this, we introduce the following restrictions:

– We consider a distributed information system with well-defined fuzzy nodes. Each node has its parameters U_k, $k = \overline{1, m}$, and m is the total number of nodes. Each parameter has a nominal value U_k^0. By observing the dynamics of these parameters, we can evaluate the changes in the processes within the system.

– During the operation of a distributed information system with fuzzy nodes, the technical parameters of its elements degrade. This can lead to a deterioration of the parameters of the elements compared to their initial values. This means that at certain points in time t_i, $i = \overline{1, n}$ the values of the determining parameters do not increase, i.e. $U_k(t_1) > U_k(t_2) > \cdots > U_k(t_n)$, if $t_1 < t_1 < \cdots < t_n$.

If the values of the parameters U_k are within the specified range of allowable values $S_k = \left(U_k^{(min)} < U_k^0 < U_k^{(max)}\right)$, where $U_k^{(min)}$, $U_k^{(max)}$ are the minimum and maximum allowable values of the parameter, respectively, then the elements function normally [14]. Deviation of the parameter value U_k from its allowable range leads to a parametric failure, which usually does not disable the system, but only degrades its performance. There is also a probability of sudden failures of vague nodes, but in this case there is a parametric failure, which can be recovered from within the framework of recovery work. This means that the distributed information system belongs to the class of highly reliable recoverable systems with a finite number of possible states [15].

We will denote the state that the system can go to at time t as $v_k(t)$, $k = \overline{0, m}$. The state of the system, when none of the m defining parameters have failed, will be denoted as $v_0(t)$. The failure of one of the system parameters will be denoted by the symbol $v_1(t)$. The symbol $v_k(t)$ describes the failure of k system parameters, and $v_m(t)$ corresponds to the failure of all m-defining parameters. The transition of the system from one state to the nearest neighboring state occurs without a sharp transition of its intermediate state. In other words, the system cannot go from state $v_0(t)$ to state $v_2(t)$ without passing through the state $v_1(t)$.

We will assume that by time t, the total cost of maintaining the system in state $v_k(t)$ is $C_k(t), k = \overline{0, m}$.

Operating costs ΔC_e per unit time are the same for each state of the system, and the average restoration costs $\Delta C_В$ are the same for each failed parameter. A critical parameter that fails at time t is restored immediately. During the restoration of the critical parameter, the system costs do not increase. Resources for restoration are allocated at the time of failure of the critical parameters.

– The transition of each fuzzy node of the system from one state to another is considered a random event [16].

We will assume that the intensity λ_k, with which the failure of parameter k of the element U_k occurs, is the same for all defining parameters.

Similar assumptions apply to the parameter μ_k, which characterizes the recovery intensity of the parameter k. This approach will allow us to calculate the total failure intensity of the system nodes, which is equal to $\lambda = \sum_{k=1}^{m} \lambda_k$ and $\mu = \sum_{k=1}^{m} \mu_k$, respectively [17].

The introduction of these restrictions requires determining the remaining system resources under conditions of limited resources for the maintenance and repair of fog nodes and their elements.

3.1. Solution

To solve the main problem, it is first necessary to calculate the resource costs for maintaining a separate information system [18] in the state $v_k(t)$, $k = \overline{0, m}$.

In $v_0(t)$, the system will be in a failure-free state until time $t + \Delta t$. This means that all nodes and their elements will be operational and functioning normally. The state of the system can be considered as the sum of two events: A_0 and B_1. These events are incompatible A_0.

Event A_0 characterizes the state of the system $v_0(t)$ at time t. It occurs when there are no failures during Δt. The probability of this event occurring is described by the equation:

$$P(A_0) = e^{-\lambda \Delta t} \approx 1 - \lambda \Delta t. \qquad (1)$$

The resources used to support the event A_0 with probability $P(A_0)$ can be calculated using the following formula:

$$C(A_0) = (C_0(t) + \Delta C_e \Delta t)(1 - \lambda \Delta t). \qquad (2)$$

If the system is in a state $v_1(t)$ and one of the parameters recovers during the time interval Δt, the event B_1 has occurred [19].

The chance of state B_1 occurring is given by the formula:

$$P(B_1) = 1 - e^{-\mu\Delta t} \approx \mu\Delta t. \quad (3)$$

The resource costs for maintaining the system in the state B_1 with probability $P(B_1)$ can be determined as follows:

$$C(\mathrm{B}_1) = C_1(t)\mu\Delta t. \quad (4)$$

As a result, the system in the state $v_0(t)$ can be considered as a sum of two incompatible events A_0 and B_1. The equation for the random function of material resource costs $C_0(t)$ on the interval $(t + \Delta t)$ to support this can be expressed as:

$$C_0(t + \Delta t) = C(A_0) + C(B_1) = \\ = (C_0(t) + \Delta C_e \Delta t)(1 - \lambda\Delta t) + C_1(t)\mu\Delta t, \quad (5)$$

or like:

$$C_0(t + \Delta t) - C_0(t) = \\ = -\lambda\Delta t C_0(t) + \mu\Delta t C_1(t) + \Delta C_e \Delta t - \lambda\Delta t \Delta C_e \Delta. \quad (6)$$

Divide both sides of equation (6) by Δt and take the limit as $\Delta t \to 0$ to obtain the differential equation:

$$\frac{dC_0(t)}{dt} = -\lambda C_0(t) + \mu C_1(t) + \Delta C_e. \quad (7)$$

For system states $v_k(t)$, $k = \overline{0, m-1}$, we write a similar equation in which the state $v_k(t)$ is characterized by the fact that at time $t + \Delta t$ the number of failures of determining parameters is equal to k.

Such a state can be considered as a sum of three incompatible events: A_k, B_{k+1}, and C_{k-1}.

If at the time t the system was in state $v_k(t)$ and during the time Δt there were no failures or restorations of any of the parameters, event A_k occurs [20]. The probability of its occurrence is determined by the expression:

$$P(A_k) = e^{-\lambda\Delta t} e^{-\mu\Delta t} \approx \\ \approx 1 - (\lambda + \mu)\Delta t. \quad (8)$$

The resource costs for maintaining the system state, which is characterized by event A_k with a probability of occurrence $P(A_k)$ can be determined as follows:

$$C(A_k) == (C_k(t) + \Delta C_e \Delta t)(1 - (\lambda + \mu)\Delta t). \quad (9)$$

The symbol B_{k+1} describes the system at time t and in state $v_{k+1}(t)$ when one parameter has recovered during time Δt. The probability of its occurrence is determined by the expression:

$$P(B_{k+1}) = 1 - e^{-\mu\Delta t} \approx \mu\Delta t. \quad (10)$$

The resource costs for maintaining the system state described by the event B_{k+1} with probability $P(B_{k+1})$ can be described by the following expression:

$$C(B_{k+1}) = C_{k+1}(t)\mu\Delta t. \quad (11)$$

The next event C_{k-1} is that at time t the system is in state $v_{k-1}(t)$ and during the time Δt there is a failure of another parameter. The probability of this event occurring is given by the expression:

$$P(\mathrm{C}_{k-1}) = 1 - e^{-\lambda\Delta t} \approx \lambda\Delta t. \quad (12)$$

The resource costs for supporting the event C_{k-1} with probability $P(\mathrm{C}_{k-1})$ can be expressed as:

$$\mathrm{C}(\mathrm{C}_{k-1}) = \mathrm{C}_{k-1}(t)\lambda\Delta t. \quad (13)$$

Therefore, the state of the system $v_k(t)$ can be considered as the sum of three incompatible events A_k, B_{k+1} and C_{k-1} and the function of material resource consumption $\mathrm{C}_k(t)$ on the interval $(t + \Delta t)$ to support this event can be expressed through:

$$\mathrm{C}_k(t + \Delta t) = C(A_k) + C(B_{k+1}) + C(C_{k-1}) \\ = C_k(t) + \Delta C_e \Delta t(1 - (\lambda + \mu)\Delta t) + C_{k+1}(t)\mu\Delta t + C_{k-1}(t)\lambda\Delta t. \quad (14)$$

Let's perform a transformation of expression (14) similar to the previous one and we will obtain the differential equation:

$$\frac{\partial C_k(t)}{\partial t} == \lambda C_{k-1}(t) - (\lambda + \mu)C_k(t) + \mu C_{k+1}(t) + \Delta C_e + \lambda\Delta C_b. \quad (15)$$

Describing the system state $v_m(t)$, when at time $t + \Delta t$ there will be a failure of all m determining parameters. Such a state can be considered as a sum of two incompatible events: A_m and C_{m-1} [21].

Event A_m characterizes the system in the state $v_m(t)$, at time t, when no parameter failures are recorded for a time change of Δt. The probability of this event is determined by the expression $P(A_m) = e^{-\mu\Delta t} \approx 1 - \mu\Delta t$, and the resource consumption for maintaining the event A_m with probability $P(A_m)$ can be expressed by the equation:

$$\mathrm{C}(A_m) == (C_m(t) + \Delta C_e \Delta t)(1 - \mu\Delta t). \quad (16)$$

Event C_{m-1} characterizes the system in the state $v_{m-1}(t)$ at time t and when another parameter failure has occurred during the time Δt. The probability of this event is determined by the equation $P(C_{m-1}) = 1 - e^{-\lambda\Delta t} = \lambda\Delta t$. The resource costs for supporting the event C_{m-1} with probability $P(C_{m-1})$ can be expressed as:

$$C(C_{m-1}) = C_{m-1}(t)\lambda\Delta t. \tag{17}$$

The system in the state $v_m(t)$ can be represented as the sum of two incompatible events, A_m and C_{m-1} [22]. The expression for the function that estimates the costs of material resources $C_m(t)$ on the interval $(t + \Delta t)$ to support this event is given by the following equation:

$$\begin{aligned} &C_m(t+\Delta t) = C(A_m) + \\ &+C(C_{m-1}) = (C_m(t) + \\ &\Delta C_e \Delta t)(1-\mu\Delta t) + (C_{m-1}(t) + \\ &\Delta C_b)\lambda\Delta t. \end{aligned} \tag{18}$$

By transforming equation (18), we obtain the following differential equation:

$$\begin{aligned} \frac{\partial C_m(t)}{\partial t} &== \lambda C_{m-1}(t) - \mu C_m(t) \\ &+ \Delta C_e + \lambda \Delta C_b. \end{aligned} \tag{19}$$

The costs of material resources to support a system in the state $v_k(t)$, $k = \overline{0,m}$ can be described by a system of equations involving unknown random functions of costs $C_k(t), k = \overline{0,m}$, which can be solved by transforming it into operator form [23]:

$$A(p)X(p) = B^{(c)}(p), \tag{20}$$

where $A(p)$ is a three-diagonal matrix of coefficients of a system of equations of dimension $(m+1) \times (m+1)$.

X(p)—vector in which the components x_k (p) are the mapping of the corresponding functions C_k (t), k = ¯(0,m) [24].

$B^{(c)}(p)$—vector of the right-hand sides of the system of equations (20):

$$X(p) = \begin{vmatrix} C_0(p) \\ C_1(p) \\ C_2(p) \\ C_3(p) \\ \dots \\ C_{m-1}(p) \\ C_m(p) \end{vmatrix}; \quad B^{(c)} = \begin{vmatrix} -\dfrac{\Delta C_e}{p} \\ -\dfrac{\Delta C_e + \lambda\Delta C_b}{p} \\ \dots \\ -\dfrac{\Delta C_e + \lambda\Delta C_b}{p} \end{vmatrix}, \tag{21}$$

p is a complex variable in the existing half-plane of the mapping functions $C_k(t), k = \overline{0,m}$.

According to [25–28], we will construct an estimate of the vector $X(p)$ of unknowns of the system of equations (20), using the following procedure for estimating the matrix coefficients of a multidimensional regression analysis model with an arbitrary finite number of regressions.

Let us take the system (20), neglecting the variable p for the sake of simplicity of the notation and introducing it into the following expression [29]:

$$\sum_{k=1}^{m+1} A_k X_k = B^{(c)}, \tag{22}$$

where:

A_k is the block components of the matrix $A = \| |A_1| \cdot |A_2| \cdot \dots \cdot |A_k| \cdot \dots \cdot |A_{m+1}| \|$.

X_k is the block components of the vector $X = \| |X_1| \cdot |X_2| \cdot \dots \cdot |X_k| \cdot \dots \cdot |X_{m+1}| \|$.

T is the transposition sign.

Using the previously introduced notation, one can find the estimate of $\dot{X}^{\cdot}$ of the vector X of unknown systems (22) from the recursive relationship:

$$\dot{X}_k = C_{k-1} = M_k A_k \prod_{\substack{j=1 \\ j\neq 1}}^{m+1} R_j B^{(c)}, \tag{23}$$

where $M_k = \left(A_k^T \prod_{j=1}^{k-1} R_j A_k\right)^{-1}$, $R_j = E - A_j M_j A_j^T \prod_{i=1}^{j-1} R_i$.

Applying the inverse Laplace transform to the estimates (23), we can obtain the expressions of the original random functions of the material costs $C_k(t)$ for the support of the operation of fog nodes in the distributed information system of a critical infrastructure object $v_k(t)$,, when at the time t from the total number of m determining parameters, some of their number $k = 0, 1, 2, \dots m$ may fail [30].

The average material resource expenditure $\bar{C}(t)$ is equal to the mathematical expectation of all possible values of the random function $C_k(t)$:

$$\bar{C}(t) = \sum_{k=0}^{m} C_k(t) P_k(t), \tag{24}$$

where $P_k(t)$ is the probability of occurrence of the event $v_k(t)$.

Equation (25) allows us to calculate the average cost of material resources required to maintain fog nodes in each of their possible states. If the average cost $\bar{C}(t=\dot{t})$ at time $t=\dot{t}$ is close to the additional resources allocated for the operation C_0, then we can take t' as the residual resource. To find t, we can solve the equation:

$$\dot{t}=\underset{t\in T}{argmin}\, e\left(C_0-\bar{C}(t)\right), \qquad (25)$$

where $e\ (C_0-\bar{C}(t))$ is the chosen measure of approximation of the quantities C_0 and $\bar{C}(t)$:

T is the set of all possible positive values of the variable t.

As a result, we will obtain expressions for the quantities $C_k(t), k=\overline{0,m}$.

4. Conclusions

This paper examines the efficiency of a distributed information system for a critical infrastructure object.

A distributed information system is a collection of fog nodes and their elements, each with its own functional purpose and physical characteristics. Fog nodes can be physical components such as gateways, switches, routers, and servers, or virtual components such as virtualized switches and virtual machines.

As fog nodes and their elements operate, they consume their technical resources. This can lead to a failure of the critical infrastructure object or its segment.

To reduce the risk of such a failure, this paper proposes a method for calculating the residual technical resource of a fog node and its elements, taking into account their possible failures.

This method can be used to predict the probability of failure of important elements of critical infrastructure objects and to predict the cost of restoring their functionality.

Further research could focus on developing a method for assessing the consequences of malicious attacks on a fog node with low technical resources.

References

[1] A Baker, S. Rashid, O. alsaif, Fog Computing: A Comprehensive Review of Architectures, Applications, and Security Challenges, NTU J Eng. Technol. 2(2) (2023). doi:10.56286/ntujet.v2i2.614.

[2] M. Iorda, et al., Fog Computing Conceptual Model, Natl. Inst. Stand. Technol., NIST Special Publication 500–325 (2018). doi: 10.6028/NIST.SP.500-325.

[3] M. Vladymyrenko, et al., Analysis of Implementation Results of the Distributed Access Control System. in: 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (2019). doi: 10.1109/picst47496.2019.9061376.

[4] P. Anakhov, et al., Protecting Objects of Critical Information Infrastructure from Wartime Cyber Attacks by Decentralizing the Telecommunications Network, in: Workshop on Cybersecurity Providing in Information and Telecommunication Systems, vol. 3550 (2023) 240–245.

[5] H. Hulak, et al., Dynamic Model of Guarantee Capacity and Cyber Security Management in the Critical Automated System, in: 2nd International Conference on Conflict Management in Global Information Networks, vol. 3530 (2023) 102–111.

[6] V. Grechaninov, et al., Decentralized Access Demarcation System Construction in Situational Center Network, in: Workshop on Cybersecurity Providing in Information and Telecommunication Systems II, vol. 3188, no. 2 (2022) 197–206.

[7] V. Buriachok, V. Sokolov, P. Skladannyi, Security Rating Metrics for Distributed Wireless Systems, in: Workshop of the 8th International Conference on "Mathematics. Information Technologies. Education:" Modern Machine Learning Technologies and Data Science, vol. 2386 (2019) 222–233.

[8] J. Vergara, J. Botero, L. Fletscher, A Comprehensive Survey on Resource Allocation Strategies in Fog/Cloud Environments, Sensors 23(9) (2023) 4413. doi: 10.3390/s23094413.

[9] A. Buttar, M. Abbas, M. Majid, Optimized Storage and Resource Management in Fog Computing Paradigm, IGI Global (2023). doi: 10.4018/978-1-6684-7348-1.ch008.

[10] G. Li, et al., Method of Resource Estimation Based on QoS in Edge Computing, Wirel. Commun. Mob. Comput. 2018 doi: 10.1155/2018/ 7308913.
[11] C. Hong, B. Varghese, Resource Management in Fog/Edge Computing: A Survey, arXiv.org (2018) 1–22.
[12] K. Kaur, A. Singh, A. Sharma, A Systematic Review on Resource Provisioning in fog Computing, Trans. Emerg. Telecommun. Technol. 34(4) (2023). doi: 10.1002/ett.4731.
[13] M. Khan, G. Nencioni, Resource Allocation in Networking and Computing Systems: A Security and Dependability Perspective, IEEE Access (2023). doi: 10.1109/access.2023.3306534.
[14] C. Gong, et al., Dynamic Resource Allocation Scheme for Mobile Edge Computing, J Supercomput 79 (2023) 17187–17207. doi: 10.1007/s11227-023-05323-y.
[15] A. Toma, S. Samara, M. Qamhieh, Edge Computing Systems: Modeling and Resource Optimization for Augmented Reality and Soft Real-time Applications, J. Netw. Syst. Manag. 79 (2023). doi: 10.1007/s10922-023-09770-z
[16] Y. Yuan, Z. Xu, Computing Resource Allocation Scheme Based on Edge Computing Under Augmented Reality Application, J. High Speed Netw. 28(3) (2022) 143–156. doi: 10.3233/jhs-220687.
[17] X. Liu, et al., A Game-Based Computing Resource Allocation Scheme of Edge Server in Vehicular Edge Computing Networks Considering Diverse Task Offloading Modes, Sensors 24(1) (2023) 69. doi: 10.3390/s24010069.
[18] H.-Y. Wei, et al., Future Trends in Fog/Edge Computing and Networking, IEEE Commun. Mag. 61(12) (2023) 38–39. doi: 10.1109/mcom.2023.10375689.
[19] S. Alizadeh, R Ghaemi, H. Monshizadeh, An Autonomous Architecture Based on Reinforcement Deep Neural Network for Resource Allocation in Cloud Computing, Comput. 106 (2023) 371–403. doi: 10.1007/s00607-023-01220-7.
[20] X. He, H. Cao, Cloud Computing-Based Sharing Platform for High-quality Teaching Resources of Higher Vocational Physical Education, eLEOT 2023: e-Learning, e-Education, and Online Training, LNICST 543 (2024) 3–16. doi: 10.1007/978-3-031-51465-4_1.
[21] R. Mahmud, R. Kotagiri, R. Buyya, Fog Computing: A Taxonomy, Survey and Future Directions, Internet of Everything (2017) 103–130. doi: 10.1007/978-981-10-5861-5_5.
[22] A. Brogi, S. Forti, A. Ibrahim, Optimising QoS-Assurance, Resource Usage and Cost of Fog Application Deployments, Cloud Comput. Serv. Sci. (2019) 168–189. doi: 10.1007/978-3-030-29193-8_9.
[23] M. Emu, P. Yan, S. Choudhury, Latency Aware VNF Deployment at Edge Devices for IoT Services: An Artificial Neural Network Based Approach, International Conference on Communications Workshops (ICC Workshops) (2020) 1–6. DOI: 10.1109/ICCWorkshops49005. 2020.9145242.
[24] H. Zhang, et al., A Hierarchical Game Framework for Resource Management in Fog Computing, IEEE Commun. Mag. 55(8) (2017) 52–57. doi: 10.1109/mcom.2017.1600896.
[25] R. Aryal, J. Altmann, Dynamic Application Deployment in Federations of Clouds and Edge Resources Using a Multiobjective Optimization AI Algorithm, 3rd International Conference on Fog and Mobile Edge Computing (FMEC) (2018) 147–154. doi: 10.1109/fmec.2018.8364057.
[26] J. Zhao, et al., Computation Offloading and Resource Allocation for Cloud Assisted Mobile Edge Computing in Vehicular Networks, IEEE Trans. Veh. Technol. 68(8) (2019) 7944–7956. doi: 10.1109/tvt.2019.2917890.
[27] C. Wang, et al., Computation Offloading and Resource Allocation in Wireless Cellular Networks with Mobile Edge Computing, IEEE Trans. Wirel. Commun. 16(8) (2017) 4924–4938. doi: 10.1109/twc.2017.2703901.
[28] N. Kherraf, et al., Optimized Provisioning of Edge Computing Resources with Heterogeneous Workload in IoT Networks, IEEE Trans. Netw. Serv. Manag. 16(2) (2019) 459–474. doi: 10.1109/tnsm.2019.2894955.

[29] T. Bahreini, D. Grosu, Efficient Placement of Multi-Component Applications in Edge Computing Systems, 2nd ACM/IEEE Symposium on Edge Computing 5 (2017) 1–11. doi: 10.1145/3132211.3134454.
[30] Y. Shcheblanin, et al., Research of Authentication Methods in Mobile Applications, in: Cybersecurity Providing in Information and Telecommunication Systems Vol. 3421. (2023) 266–271.