UDC 004.7 **doi: 10.26906/SUNZ.2023.4.129**

Yulia Andrusenko, Tetiana Fesenko

Kharkiv National University of Radio Electronics, Kharkiv, Ukraine

INSTABILITY OF CLOUD INFRASTRUCTURE RESOURCES AND SERVICES

Abstract. Increasing non-stationarity of cloud infrastructure resources and services leads to a significant decrease in its productivity. Therefore, **the purpose of the article** is to identify the reasons for non-stationarity of cloud infrastructure resources and services; finding ways to reduce the level of non-stationarity . As a result of the research, the following **results were obtained** . Sources leading to the specified non-stationarity are identified . Existing approaches to reducing non-stationarity are analyzed . An example of basic resource allocation using standard linear programming methods is given. Variants of application of these methods for dynamic redistribution of resources are shown. **Conclusion.** Allocation and dynamic redistribution of resources in the cloud infrastructure can be done using standard linear programming methods. But due to the significant non-stationarity of the cloud environment, the proposed approach will reduce the productivity of cloud resources. At the same time, with an increase in the number of variables and restrictions, the computational complexity of the proposed algorithm will grow exponentially . Therefore, it is necessary to look for other approaches for the distribution and redistribution of cloud resources in conditions of significant non-stationarity .

Keywords: cloud environment, cloud resources, non-stationarity , uncertainty, redistribution of resources.

Introduction

Cloud computing is widely recognized by both practitioners and researchers as a reliable solution for storing and processing data for both commercial and scientific purposes. Although cloud computing has many advantages, not all problems with it have been fully solved, especially in the field of security, reliability, performance, etc. The vast majority of research in the field of planning assumes the availability of complete and reliable information about the problem, work, that is, a static deterministic execution environment. However, in cloud computing, services and resources are subject to significant non-stationarity when accessing and using resources. Volatility is a significant problem in cloud computing, creating additional challenges for end users, resource providers, and administrators. It is necessary to abandon the usual paradigms, adapt the existing models to the evolution of computing tools, and develop new strategies for planning and managing resources to effectively overcome non-stationarity . Uncertainty in user perception of qualities, intentions and actions of providers, privacy, security, availability, etc., among other aspects of cloud computing, are actively being researched [1–10]. However, their role in resource and service planning has not yet been adequately addressed.

The purpose of the article is to identify the reasons for non-stationarity of cloud infrastructure resources and services; finding ways to reduce the level of non-stationarity .

1. Sources of uncertainty

There are many sources of uncertainty. Table 1 describes some of them and briefly explains their planning implications. Among them: elasticity of work, dynamic change of characteristics, virtualization with a weak connection of applications with the infrastructure on which they are executed, variability of the time of provision of resources, inaccuracy of estimation of the time of execution of work, change of time of processing and transfer of data, time limits of processing, directive deadlines, change in real bandwidth and other phenomena. The workload can change dramatically. It is difficult to accurately estimate the time of execution of works, to build models of its forecasting taking into account the history of calculations, to dynamically adjust the forecast, to correct errors in forecasts, etc. building [11, 12].

Actual performance can vary by sharing shared resources between different virtual machines. It is impossible to gain accurate knowledge about the system. Parameters such as the actual processor speed at which the virtual machine is processed, the number of available processors, or the actual bandwidth change over time. The principle of elasticity of cloud resources, when the user can change requests for resources, has a positive effect on the quality of service, but adds a new factor of uncertainty. Uncertainty can be in various components of computing and communication processes. The following knowledge is important: how specific dynamic computing and communication characteristics affect planning; how these characteristics can be used by the scheduler to reduce the impact of information incompleteness and achieve the desired QoS ; how to effectively solve the relevant optimization problem; how to ensure scalable and reliable GRID behavior under constraints such as budget, QoS , SLA, energy cost, etc. Various stochastic, adaptive, reactive planning algorithms that do not require knowledge are studied, which are considered as effective alternatives to known technologies of deterministic optimization of the theory of schedules.

2. Approaches to reducing non-stationarity

Unspecified factors that can reduce the reliability and security of data, such as unexpected and unauthorized data modifications, hardware and software failures , disk errors, integrity violations or data loss, malicious intrusions, falsifications, denial of access for a long time, information leakage, conspiracy, etc., are considered in [13]. The authors propose a customizable, reliable, and secure distributed data storage scheme with an improved data detection and correction method, as well as their encoding/decoding speed. Technical failures, data breaches and collusion are difficult to predict.

Table 1 – **Cloud infrastructure parameters and their main sources of uncertainty**

	Data	**Virtualization**	**Receipt of works**	**Migration**	**Energy consumption**	**Scalability**	**Availability of resources**	**Traffic elasticity**	**Elastic provision of resources**	**Time to provide resources**
Real performance		+	+	+	+		+	+	+	+
Real bandwidth	+		+	+	+	+	+	+	+	+
Processing time		+			+		+	+	+	+
Available memory	+				+	+	+	+	+	+
Number of processors		+			+	+	+	+	+	+
Available storage space	+				+	+	+	+	+	+
Data transfer time	+								+	+
Resource potential		+	+			+		+	+	+
Network performance	+			+		+			+	+

This type of uncertainty is one of the main challenges when designing a reliable IT infrastructure. Modified Asmuth-B thresholds and weighted Mignotte secret sharing schemes are proposed to reduce encryption redundancy and damage caused by cloud collusion [14].

Technical failures, data breaches and collusion are difficult to predict. This type of uncertainty is one of the main challenges when designing a reliable IT infrastructure.

Most existing optimization solutions assume that the behavior of virtual machines and services is predictable and stable in operation. In real cloud infrastructures, these assumptions are not justified. Although most providers guarantee a certain processor speed, memory capacity, and amount of local storage for each dedicated virtual machine, the actual performance depends on the physical hardware used, as well as the use of shared resources by other virtual machines assigned to the same computer. This is also true for the communication infrastructure, where the actual bandwidth is very dynamic and difficult to guarantee.

The combination of virtualized computing resources, storage, software and services of dynamically scaling cloud environments adds a new dimension to the planning problem. The way services are provided depends not only on the properties and required resources, but also on other users who share the resources. Managing and optimizing cloud infrastructure is a complex task. Existing planning models do not accurately account for the uncertainty and dynamic performance changes inherent in heterogeneous and distributed infrastructures .

Two types of uncertainty can be distinguished: parametric and systemic. Parametric uncertainties arise as a result of incomplete knowledge and changes in parameters, for example, when data are imprecise, they are estimated using statistical methods and expressed as probabilities. Their analysis quantifies the impact of random input variables on optimization results. The efficiency and accuracy of probabilistic uncertainty analysis is a matter of trade-off. This type of uncertainty is not reduced because it is a property of the system itself. Systemic uncertainty arises as a result of incomplete understanding of the processes that control service planning and can be reduced by obtaining more information. It is modeled using the theory of probabilities, the theory of evidence, the theory of possibilities, and fuzzy sets.

A robust design minimizes the impact of uncertainties on system performance and behavior. Traditionally, it was carried out using either a probabilistic approach or a worst-case analysis. Both approaches treat uncertainty either as random variables or as interval variables. In fact, uncertainty can be a combination of both.

Understanding and accounting for uncertainty should lead to improved resource planning efficiency. Most cloud applications require communication resources to exchange information between services, databases, or end users. However, providers may know the amount of data they will have to manage or the amount of computation required to perform the work.

One of the possible approaches to reducing non-stationarity is to allocate resources using standard methods. After that, with increasing non -stationarity , it is suggested to redistribute resources. Let's consider this approach using a simple example.

3. An example of a basic distribution of resources

Two streams of Big Data (stream A , traffic T_A terabytes per hour; stream B , traffic T_B terabytes per hour) are processed in the cloud environment. The corresponding cloud infrastructure has two critical resources during the current maintenance cycle. The first resource is dedicated RAM for storing the generalized results of data processing . The existing limitation is a_1 Mb . The second resource is fast external memory for storing selected data for further use. The existing limitation is a_2 GB .

On average, in one hour, the flows have the following characteristics:

stream *A* summarizes *a* $_{1A}$ MB of information and selects for storage *a* $_{2A}$ GB of data;

flow *B* summarizes *a* $_{1B}$ MB of information and selects for storage *a* $_{2V}$ GB of data.

We will consider the example on the following specific data:

traffic intensity of information flows: T $_{A}$ =5, T $_{B}$ =4 terabytes per hour;

the amount of allocated RAM: and $_1$ = 24 MB ;

the size of the allocated fast external memory: and $_2$ = 6 GB ;

intensity of receipt of generalized data reprocessing results : a $_{1A}$ = 6 Mb /h.; and $_{1V}$ = 4 Mb /h;

the intensity of receiving the selected data: a $_{2A}$ = 1 Gb /h.; and $_{2V}$ = 2 GB / hour.

In addition, during the current service cycle, the cloud infrastructure can process no more than 8 terabytes of stream *B* , that is, data processing takes no more than 8/4=2 (hours). Also, the duration of service of stream *B* cannot exceed the duration of service of stream *A* by more than 1 hour.

It is necessary to find the optimal time of the current service cycle, based on the requirement of the maximum possible total traffic received for processing from both streams.

First, it is necessary to carry out a mathematical formalization of this task. To do this, we will introduce the variables *x* $_{A}$ and *x* $_{B}$ - the duration of service of Big Data flows in the cloud environment (in hours).

Based on the verbal constraints described above, the system of task constraints can be formalized as follows:

$$x_A \cdot 6 + x_B \cdot 4 \le 24; \qquad (1)$$

$$x_A \cdot 1 + x_B \cdot 2 \le 6; \qquad (2)$$

$$x_B - x_A \le 1; \qquad (3)$$

$$x_B \le 2. \qquad (4)$$

In addition, there are implicit semantic restrictions (the value of the duration of maintenance of flows on a switching node cannot be negative):

$$x_A \ge 0; \qquad (5)$$

$$x_B \ge 0. \qquad (6)$$

The total volume of traffic for one processing cycle is equal to the sum of 5 *x* $_{A}$ + *4* x $_{B.}$ Therefore, the objective function of the task will look like this:

$$y = 5x_A + 4x_B \to \max\ . \qquad (7)$$

Therefore, the standard form of the linear programming problem is obtained (expressions (1) - (7)). For its solution, with a small number of constraints and variables, any existing methods can be used, for example, the simplex method.

For clarity, we will find the values of the required variables using the graphical method (we will use the presence of only two variables).

In the xA OxБ coordinate system, we successively construct a polygon of permissible values (Figs. 1-3), based on the formalized constraints (1) - (6) .

The points of the space of admissible solutions marked in Fig. 4 in blue, satisfy all restrictions at the same time. This space is limited by straight line segments that connect at the vertices of the resulting polygon.

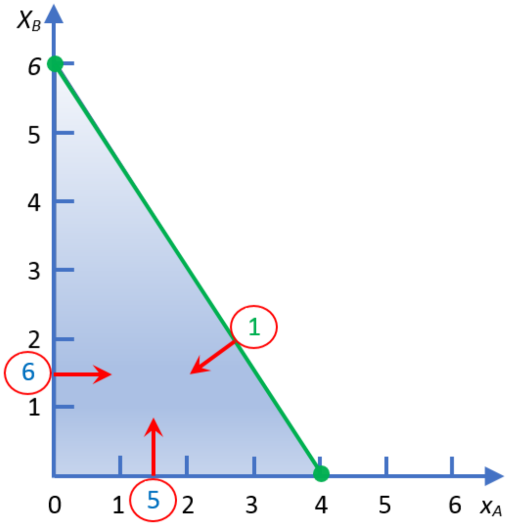

Fig. 1. Consideration of implicit restrictions and limitations (1)

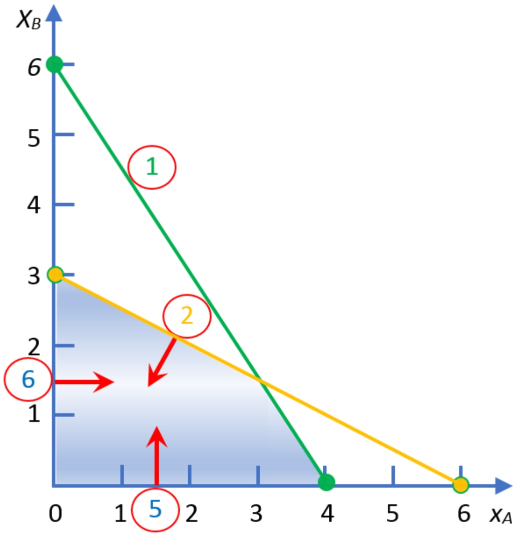

Fig. 2. Adding a restriction (2)

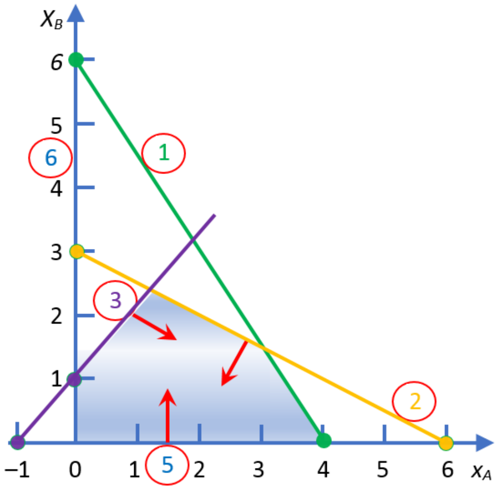

Fig. 3. Adding a restriction (3)

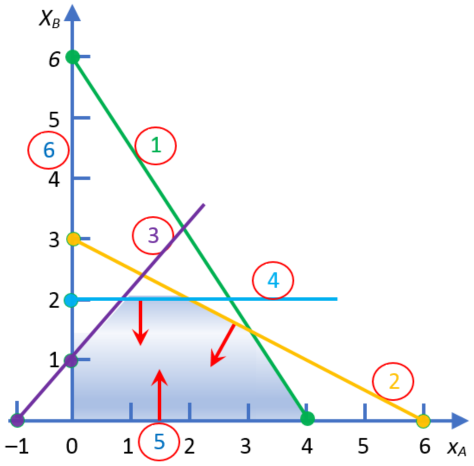

Fig. 4. Polygon of permissible values

Any point located inside or on the boundary of a given polygon is an admissible solution, that is, it satisfies all constraints. Since the feasible solution space contains an infinite number of points, some procedure for finding the optimal solution is required.

In order to find the optimal solution, it is necessary to determine the growth direction of the objective function (7) (since it should be maximized). We can equate z to several increasing values. These values, substituted instead of z in the expression of the objective function, generate the equations of straight lines (Fig. 5). The objective function can increase as long as the straight lines corresponding to the increasing values of this function cross the region of admissible solutions. The point of intersection of the area of admissible solutions and the line corresponding to the maximum possible value of the objective function will be the point of the optimum.

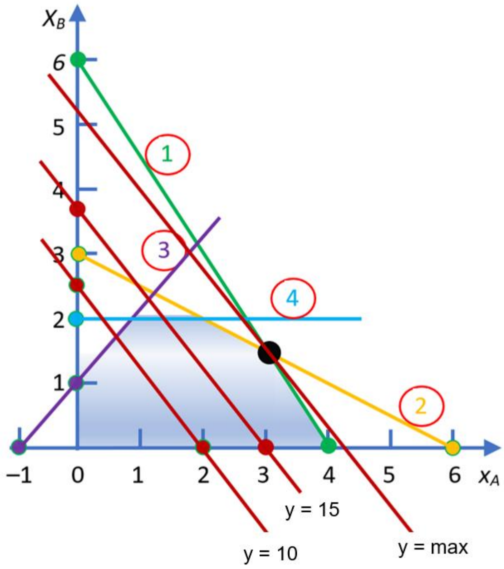

Fig. 5. Projection of the objective function

In fig. 5, it can be seen that the optimal solution corresponds to the point highlighted in black. This point is the intersection of straight lines (1) and (2), so its coordinates x A and x B are found as solutions to the system of equations defining these straight lines:

$$\begin{cases} 6x_A + 4x_B = 24; \\ x_A + 2x_B = 6. \end{cases}$$

The solution of this system will be $x_A = 3$ and $x_B = 1.5$, while the value of the objective function is z = 21.

Therefore, the required time of the current service cycle will be three hours, while reprocessing will be performed for 21 terabytes of Big Data.

4. Dynamic redistribution of resources

In the example of point 3, the result is obtained for unchanged task data. But in the process of processing, any changes may occur. Some restriction can be either strengthened or completely removed. Data acquisition rates may also change. Such situations can be resolved by varying the task statement.

For example, consider the option of removing restriction (4). As we can see in fig. 6, the result remained unchanged.

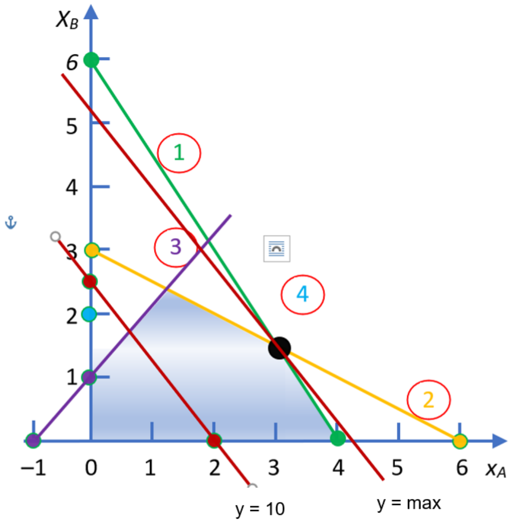

Fig. 6. Removal of restriction (6)

But removing a significant constraint or changing the objective function may change the result. This can be seen in Fig. 7, where restriction (1) is removed. As a result, it turned out to be expedient to rework only flow *A* , and the time of the current service cycle increases.

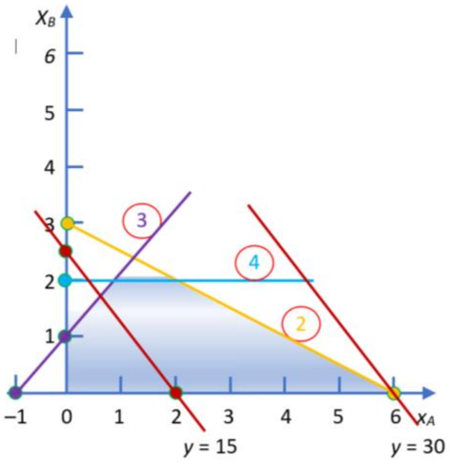

Fig. 7. Removal of restriction (1)

But it should be noted that due to the non-stationarity of cloud infrastructure resources and services, this step will have to be performed very often.

Conclusion

Allocation and dynamic redistribution of resources in the cloud infrastructure can be done using standard linear programming methods. But due to the significant non-stationarity of the cloud environment, the proposed approach will reduce the productivity of cloud resources. At the same time, with an increase in the number of variables and restrictions, the computational complexity of the proposed algorithm will grow exponentially.

Therefore, it is necessary to look for other approaches for the distribution and redistribution of cloud resources in conditions of significant non-stationarity.

References

1. Mezni, H., Aridhi, S.and Hadjali, A. (2018), “The uncertain cloud: State of the art and research challenges”, *International Journal of Approximate Reasoning*, Vol. 103, pp. 139-151, doi: https://doi.org/10.1016/j.ijar.2018.09.009.
2. Nawrocki, P., Grzywacz, M. and Sniezynski, B. (2021), “Adaptive resource planning for cloud-based services using machine learning”, *Journal of Parallel and Distributed Computing*, Vol. 152, pp. 88-97, doi: https://doi.org/10.1016/j.jpdc.2021.02.018.
3. Saidi, K., Hioual, O. and Siam, A. (2020), “Resources Allocation in Cloud Computing: A Survey”, *ICAIRES 2019: Smart Energy Empowerment in Smart and Resilient Cities*”, pp 356–364, doi: https://doi.org/10.1007/978-3-030-37207-1_37.
4. Habiba, U., Masood, R., Shibli, M.A. and Niazi, M.A. (2014), “Cloud identity management security issues & solutions: a taxonomy”, *Complex Adapt Syst Model*, Vol. 2, 5, doi: https://doi.org/10.1186/s40294-014-0005-9.
5. Kuchuk, N., Mozhaiev, O., Semenov, S., Haichenko, A., Kuchuk, H., Tiulieniev, S., Mozhaiev, M., Davydov, V., Brusakova, O. and Gnusov, Y. (2023), “Devising a method for balancing the load on a territorially distributed foggy environment”, *Eastern-European Journal of Enterprise Technologies*, 1(4 (121), pp. 48–55. doi: https://doi.org/10.15587/1729-4061.2023.274177.
6. Chen, J., Wang, Y. and Liu, T. (2021), “A proactive resource allocation method based on adaptive prediction of resource requests in cloud computing”, *J. Wireless Com Network*, 24, doi: https://doi.org/10.1186/s13638-021-01912-8.
7. Petrovska, I. and Kuchuk, H. (2022), “Static allocation method in a cloud environment with a service model IAAS”, *Advanced Information Systems*, vol. *6, is.* 3, pp. 99–106, doi: https://doi.org/10.20998/2522-9052.2022.3.13.
8. Kuchuk, N., Shefer, O., Cherneva, G. and Alnaeri, F.A. (2021), “Determining the capacity of the self-healing network segment”, *Advanced Information Systems*, vol. 5, no. 2, pp. 114–119, Jun. 2021, doi: https://doi.org/10.20998/2522-9052.2021.2.16.
9. Petrovska, I. and Kuchuk H. (2022), “Features of the distribution of computing resources in cloud systems”, *Control, Navigation and Communication Systems*, No. 2, pp. 75-78, doi: http://dx.doi.org/10.26906/SUNZ.2022.2.075.
10. Kuchuk, G., Kovalenko, A., Komari, I.E., Svyrydov, A. and Kharchenko, V. (2019), “Improving big data centers energy efficiency: Traffic based model and method”, *Studies in Systems, Decision and Control*, vol. 171, Kharchenko, V., Kondratenko, Y., Kacprzyk, J. (Eds.), Springer Nature Switzerland AG, pp. 161-183, doi: https://doi.org/10.1007/978-3-030-00253-4_8.
11. Nechausov, A., Mamusuĉ, I. and Kuchuk, N. (2017), “Synthesis of the air pollution level control system on the basis of hyperconvergent infrastructures”, *Advanced Information Systems*, vol. 1, no. 2, , pp. 21–26, doi: https://doi.org/10.20998/2522-9052.2017.2.04.
12. Tan, B., Ma, H. and Mei, Y. (2017), “A NSGA-II-based approach for service resource allocation in cloud”, *IEEE Congress on Evolutionary Computation (CEC)*, 17013723, pp. 2574–2581, doi: https://doi.org/10.1109/CEC.2017.7969618.
13. Mohamed, Abdel-Basset, Laila, Abdel-Fatah and Arun Kumar Sangaiah. (2018), “Chapter 10 - Metaheuristic Algorithms: A Comprehensive Review”, Editor(s): Arun Kumar Sangaiah, Michael Sheng, Zhiyong Zhang, *Intelligent Data-Centric Systems*, Computational Intelligence for Multimedia Big Data on the Cloud with Engineering Applications, Academic Press, 2018, Pages 185-231, ISBN 9780128133149, doi: https://doi.org/10.1016/B978-0-12-813314-9.00010-4.
14. Liu, Xi, and Dan Zhang. 2019. "An Improved SPEA2 Algorithm with Local Search for Multi-Objective Investment Decision-Making" *Applied Sciences* 9, no. 8: 1675. doi: https://doi.org/10.3390/app9081675.

Received (Надійшла) 18.09.2023
Accepted for publication (Прийнята до друку) 15.11.2023

Нестаціонарність ресурсів та послуг хмарної інфраструктури

Ю. О. Андрусенко, Т. Г. Фесенко

Анотація. Збільшення нестаціонарності ресурсів та послуг хмарної інфраструктури призводить до суттєвого зниження її продуктивності. Тому **метою статті** є визначення причин нестаціонарності ресурсів та послуг хмарної інфраструктури; знаходження шляхів зменшення рівня нестаціонарності. В результаті дослідження отримані такі **результати**. Визначені джерела, що призводять до вказаної нестаціонарності. Проаналізовані існуючі підходи до зменшення нестаціонарності. Наведений приклад базового розподілу ресурсів з використанням стандартних методів лінійного програмування. Показані варіанти застосування даних методів для динамічного перерозподілу ресурсів. **Висновок.** Розподіл та динамічний перерозподіл ресурсів у хмарній інфраструктурі можна провести з використанням стандартних методів лінійного програмування. Але за рахунок суттєвої нестаціонарності хмарного середовища запропонований підхід знизить продуктивність хмарних ресурсів. При цьому при збільшенні кількості змінних та обмежень обчислювальна складність запропонованого алгоритму буде зростати експоненційно. Отже, необхідно шукати інші підходи для розподілу та перерозподілу хмарних ресурсів в умовах суттєвої нестаціонарності.

Ключові слова: хмарне середовище, хмарні ресурси, нестаціонарність, невизначеність, перерозподіл ресурсів.