

NINA KUCHUK<sup>1</sup>, VIKTORIIA MERLAK<sup>2</sup><sup>1</sup> National Technical University “Kharkiv Polytechnic Institute”, Ukraine<sup>2</sup> National Aerospace University “Kharkiv Aviation Institute”, Ukraine

## MODELING OF INFORMATION INTERCONNECTION IN A COMPUTER CONTROL SYSTEM OF MOVING OBJECTS

*The subject of research* is a pool of widgets, which provides a user interface in a computer control system for moving objects. *The object of research* is the process of information exchange between the components of a computer system, which are directly involved in the control of moving objects. The task of allocating resources for a pool of widgets that provide operational control of moving objects is critical. As the number of elements increases in the pool of widgets, the competition for the resources of the computer system increases significantly, respectively, the queues for the most demanded resources grow. Therefore, it is necessary to reallocate resources between requests in such a way that the priorities of requests are taken into account, and the response time of operational requests does not exceed the standard time. Such distribution cannot be realized without a model that would show the parameters of all information flows between the components, which is **the purpose of this article**. In doing so, the model must consider the characteristics of the pool of widgets that support the user interface. **Research results.** To achieve this goal, several particular tasks were solved. First, the main sets were identified that will be involved in modeling information interconnections between software components that service requests for user interface widgets in a computer system environment. Then, a diagram of informational interconnections between these sets was developed. Further, the parameters of the requests, which were initiated by the widgets, of the applications, and the fragments of the widget query service database were determined and formalized. All this made it possible to form a model of information interconnections using a tuple of obtained vectors, sets, and Boolean matrices. In the proposed mathematical model, many limitations are imposed on the elements of this tuple, associated with the characteristic features of the control module for the user interface of the computer system. Besides, the limitations associated with the functioning of the basic network of the computer system are taken into account. In the final model, Boolean matrices affecting resource allocation defined as model variables. A fast algorithm for determining the space of admissible distributions is proposed. **Conclusions.** The developed mathematical model determines the parameters of information flows between the components of the computer system, which are involved in the control of moving objects. This model will allow for the operational reallocation of resources between requests using the proposed fast algorithm.

**Keywords:** pool of widgets; user interface; moving object; computer system; information interconnections.

### Introduction

**Relevance.** Nowadays, computer systems of critical application are getting more and more development. At the same time, at present, considerable attention is paid to the interface of the end users of the system [1]. Typically, this interface is implemented using an appropriate pool of widgets.

Depending on the purpose of the widget pool item, there may be different requirements for it. In particular, for widgets that provide a user interface to a component of a computer system, the execution time of a request initiated by a particular widget is critical. In turn, this indicator depends on the efficiency of the system during providing the required resources for the request. Therefore, the task of allocating resources for a pool of widgets that provide operational control of mobile objects is important.

**Motivation.** As the number of elements in the widget pool increases, the competition for the resources of the computer system increases. This is especially noticeable when controlling moving objects using a special user interface. For example, consider a computer system that controls a group of unmanned aerial vehicles (UAVs) to conduct a post-emergency survey. The standard response time to a request initiated by the pool widget that provides the user interface of this system –  $T_{spec}$ . The dependence of the average response time to a request ( $T_{average}$ ) on the number of UAVs in the considered group ( $N$ ) is shown in Fig. 1.

It can be seen from the graph that with an increase in the number of UAVs in a grouping, the response time to a request also increases significantly. Note that increasing the value  $N$  increases the size of the widget pool.

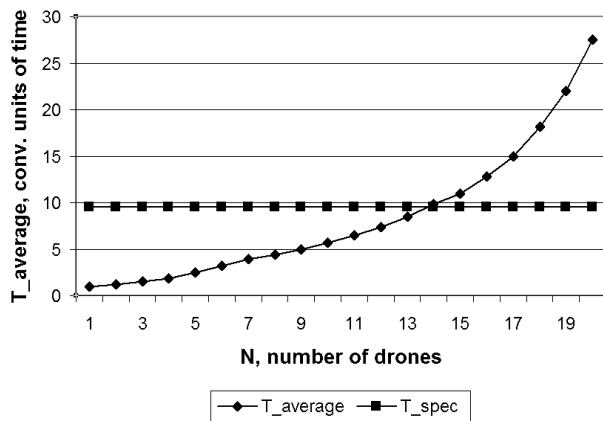


Fig. 1. Average response time to request

At the same time, the main reason for delays is the queues to the most demanded resources. Therefore, it is necessary to reallocate resources between requests in such a way that the priorities of requests are taken into account, and the response time of operational requests does not exceed the standard time  $T_{spec}$ . Such a distribution cannot be realized without a model showing the parameters of all information flows within a given fragment of the network. In doing so, the model must take into account the characteristic features of the widget pool.

**Work related analysis.** The issues of optimal allocation of resources of computer systems are considered in many scientific studies. So, in [2], the query processing speed is increased due to processing parallelization. However, when working with a pool of widgets, in most cases this is impossible to implement. In [3], the author proposes a random swap clustering algorithm. However, it significantly loses its effectiveness with an increase in the number of moving objects. When synthesizing the structure of a computer system in [4], the specificity of the individual components of the system is not considered. The methods proposed in [5, 6] are effective only for mobile components. The articles [7, 8] take into account to a greater extent the features of moving objects, rather than the characteristics of a computer system. Articles [9-11] focus on security issues in resource allocation, overshadowing time indicators. The emphasis in works [12-14] is made on the specifics of Big Data, which is unusual for a pool of widgets. In articles [15, 16], only a separate class of computer systems is considered, which has a hyperconverged platform. The article [17] focuses only on the choice of a database management system. And in article [18] the emphasis is made only on composite applications.

A common drawback of all the considered approaches to the development of a model of information interconnections is the lack of consideration of the specifics of the functioning of the pool of widgets that sup-

port the user interface. This leads to suboptimal distribution of system resources between requests for the corresponding widgets. The consequence of this is an unjustified increase in the response time of the system to a request.

**The aim of this paper** is the development of a mathematical model of information interconnections in a computer system that is responsible for the control of moving objects. The model should be focused on the features of the functioning of the pool of widgets that support the user interface.

To achieve this goal, it is necessary to solve the following tasks:

- 1) development of a scheme of information interconnections in the environment of a computer system to control moving objects;
- 2) development of a mathematical model describing information interconnections in the control of moving objects;
- 3) determination of the space of admissible distributions over the nodes of the computer system of software elements in the control of moving objects;
- 4) analysis of the results.

The article consists of four main sections.

1. Mathematical model of information interconnections.
2. Defining the space of admissible distributions.
3. Case study.
4. Application perspectives.

## 1. Mathematical model of information interconnections

The user interface management module is responsible for coordinating widget requests. It receives requests from users of the computer system, which are formed by the corresponding widgets, and launches the necessary system applications. The application data set consists of two subsets: applications that work directly with the supervisor of the basic network (Z1), and applications that communicate with a distributed database of a computer system (Z2).

Composite applications, depending on the incoming request, can perform different functions. Therefore, each function of such an application will be considered in the model as a separate application.

Let's designate the main sets that will be involved in modeling information interconnections between software components that serve requests for user interface widgets in a computer system environment:

$M_V$  – a set of widgets that make up the user dashboard widget pool,  $\dim M_V = V$ ;

$M_N$  – a set of nodes of a computer system on which applications and fragments of the database for

servicing widget requests are located,  $\dim M_N = N$ ;

$M_Z$  – a set of applications that serve a pool of widgets,  $\dim M_Z = Z$ ;

$M_Q$  – a set of user(s) requests of a computer system,  $\dim M_Q = Q$ ;

$M_D$  – a set of fragments of database serving widget requests,  $\dim M_D = D$ .

It should be noted that many applications  $M_Z$  regarding the purpose of processing requests for widgets can be conditionally divided into two such subsets:

$$M_Z = M_{Z1} \cup M_{Z2}.$$

The diagram of information interconnections in the environment of a computer system is shown in the Fig. 2.

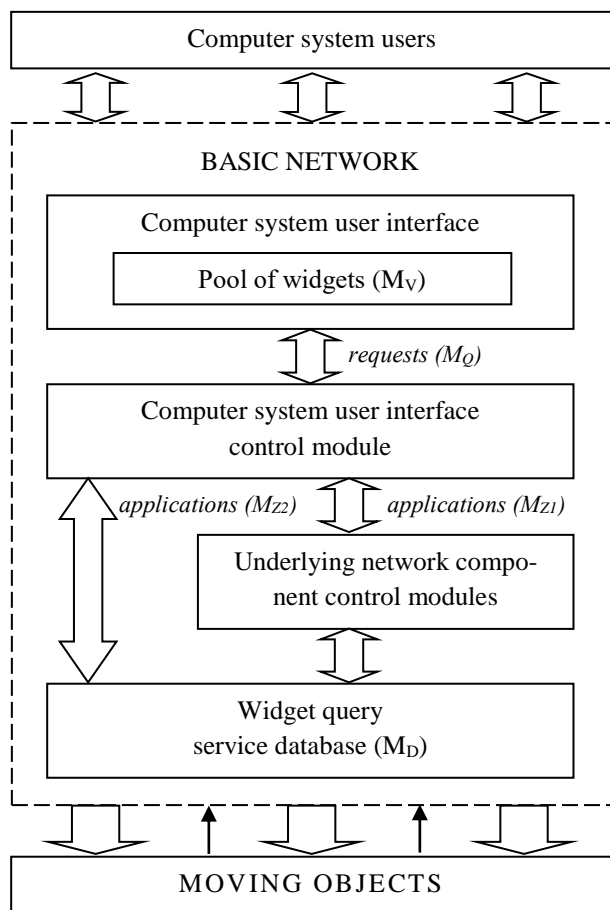


Fig. 2. Information interconnections

The parameters of each request  $q \in M_Q$  ( $q = \overline{1, Q}$ ) are defined by a tuple

$$Q_q = \langle Z_q, D_q, V_q, W_q \rangle, \quad (1)$$

where  $Z_q = (z_{qz}) = (z_{q1}, \dots, z_{qZ})$  – the vector of those

applications that are needed to execute the query  $q$ ; accordingly, vector  $D_q = (d_{qd}) = (d_{q1}, \dots, d_{qD})$  specifies the necessary fragments of the database for servicing widget requests, and  $V_q = (v_{qv}) = (v_{q1}, \dots, v_{qV})$  – the vector of widgets that require launching a query  $q$ , and the sequence of launching applications for this request is determined by the matrix  $W_q = (w_{ij})$ ,  $i, j = \overline{1, Z}$ .

It should be noted that all considered vectors and the matrix are Boolean, and take the value of one when the corresponding conditions are met, when the query  $q$  uses the corresponding resource.

Parameters for application  $z$  request  $q$  are set by a tuple

$$Z_{zq} = \langle \bar{\lambda}_{zq}, \bar{\beta}_{zq} \rangle, \quad (2)$$

where vector  $\bar{\lambda}_{zq} = (\lambda_{zq1}, \dots, \lambda_{zqD})$  defines the amount of data that application  $z$  (activated by request  $q$ ) must transfer or retrieve from the corresponding fragment of the database; vector  $\bar{\beta}_{zq} = (\beta_{zq1}, \dots, \beta_{zqZ})$  defines the amount of data that is required to exchange with other applications when executing query  $q$ .

The placement of applications on the nodes of a computer system is specified by a rectangular Boolean matrix of size  $Z \times N$ :

$$G = (g_{zn}), \quad (3)$$

where  $g_{zn} = 1$ , if this application  $z$  is launched from information node  $n$ ,  $z \in \overline{1, Z}$ ,  $n \in \overline{1, N}$ .

The placement of the software, which initiates the launch of the corresponding requests when the widget is activated, is assigned to the nodes of the computer system by a rectangular Boolean matrix with a size of  $V \times N$ :

$$H = (h_{vn}), \quad (4)$$

where  $h_{vn} = 1$  then, if for the software of the widget with number  $v$ , the scheduled information node is  $n$ ,  $v \in \overline{1, V}$ ,  $n \in \overline{1, N}$ .

The placement of fragments of the database for servicing requests for widgets on the nodes of a computer system is specified by a rectangular Boolean matrix of size  $D \times N$ :

$$S = (s_{dn}), \quad (5)$$

where  $s_{dn} = 1$  then, if the fragment of the database with the number  $d$  will be located on the node of the computer system  $n$ ,  $d \in \overline{1, D}$ ,  $n \in \overline{1, N}$ .

Thus, it is possible to form a model of information interconnections using such a tuple of the above vectors, sets and zero matrices:

$$\mathfrak{S} = \left\langle \begin{array}{l} M_V, M_N, M_Z, M_Q, \{Z_{zq} = \langle \bar{\lambda}_{zq}, \bar{\beta}_{zq} \rangle\}, \\ M_D, G, H, S, \{Q_q = \langle Z_q, D_q, V_q, W_q \rangle\} \end{array} \right\rangle. \quad (6)$$

But the module for controlling the user interface of a computer system imposes a number of restrictions on the elements of this tuple. In addition, the limitations associated with the operation of the basic network of the computer system should be considered. So:

– any request must initiate at least one computer system application, therefore

$$\sum_{i=1}^Z z_{qi} \geq 1 \quad \forall e \in \overline{1, E}; \quad (7)$$

– there are requests that appeal directly to the modules for managing the components of the basic network without exchanging with the widget request service database, that is

$$\sum_{i=1}^D d_{qi} \geq 1 \quad \forall q \in \overline{1, Q}; \quad (8)$$

– each request can be initiated by at least one widget, therefore

$$\sum_{i=1}^V u_{qi} \geq 1 \quad \forall q \in \overline{1, Q}; \quad (9)$$

– each widget, upon activation, must initiate at least one request, that is

$$\sum_{i=1}^Q v_{iv} \geq 1 \quad \forall v \in \overline{1, V}; \quad (10)$$

– each application from set  $Z$  must be installed on only one node (all components of composite applications are considered as separate elements of this set):

$$\sum_{i=1}^N g_{zi} = 1 \quad \forall z \in \overline{1, Z}; \quad (11)$$

– during the modeling considers (included in the set of nodes) only those nodes of the computer system that are involved in processing of requests of the pool of widgets, therefore

$$\sum_{i=1}^Z g_{in} + \sum_{j=1}^Q g_{jn} \geq 1 \quad \forall n \in \overline{1, N}; \quad (12)$$

– at the time the widget pool starts functioning, all applications that are needed to process widget requests must be installed, that is

$$\sum_{i=1}^Z \sum_{j=1}^N g_{ij} = Z; \quad (13)$$

– software for activating widgets must be distributed over the nodes of the computer system (if a node serves several widgets, then such a node is decomposed), so

$$\sum_{i=1}^V \sum_{j=1}^N h_{ij} = N; \quad (14)$$

– each widget is assigned to only one node of the computer system, that is

$$\sum_{i=1}^N h_{vi} = 1 \quad \forall v \in \overline{1, V}; \quad (15)$$

– pinning widgets to nodes is arbitrary, therefore

$$\sum_{i=1}^N h_{in} \in [0, V]; \quad (16)$$

– all fragments of the database for servicing requests for widgets must be distributed among the nodes of the computer system, that is

$$\sum_{i=1}^D \sum_{j=1}^N s_{ij} \geq D \quad \forall d \in \overline{1, D}; \quad (17)$$

– data replication is possible, which means that one fragment of the widget query service database can be located on several nodes, that is

$$\sum_{i=1}^N s_{di} \geq 1; \quad (18)$$

– the allocation of fragments of the data storage on the nodes of the computer system is arbitrary, that is

$$\sum_{i=1}^D s_{in} \in [0, D] \quad \forall i \in \overline{1, N}. \quad (19)$$

So, the tuple (6), considered in conjunction with conditions (7) - (19), is a mathematical model of information interconnections in the operation of the software of the pool of widgets in the environment of a computer system.

## 2. Defining the space of admissible distributions

Boolean matrices  $G, H, S$  in the mathematical model (6) - (19) set the location of the software, initiate the launch of the corresponding queries when activating the widget, applications and database fragments. Also, Boolean matrices are responsible for servicing widget requests for the nodes of the computer system. It is obvious that the quality of the functioning of the control module for the user interface of a computer system significantly depends on these matrices, that is, in a given mathematical model, Boolean matrices  $G, H, S$  can be considered model variables. Then the number of possible distribution options is  $2^{(Z+V+D) \times N}$ . Even with a small number of computer system nodes and widgets, as well as with a minimal database distribution, the number of possible options is very large.

Note that many separate widgets are usually used to control moving objects. Therefore, when using the developed mathematical model to find the best options for the location of components, it is necessary to significantly reduce the number of options under consideration, and as quickly as possible. For this purpose, it is necessary to use the features of Boolean matrices

G, H, S and some of the limitations (7) - (19).

Rectangular Boolean matrices have the same number of columns. It allows combining these matrices and considers in the future only one, combined matrix, which specifies the variant of the distribution of components over the nodes of the computer system:

$$\begin{aligned} G_{com} &= G \vee H \vee S; \\ \text{card}(G_{com}) &= (Z + V + D) \times N. \end{aligned} \quad (20)$$

The next step is to analyze how the above limitations of the mathematical model affect the existing matrix  $G_{com}$ .

Requests initiated by widgets are always launched from the node of the computer system on which installed the corresponding widget software. Therefore, limitations on queries (7) - (10) do not affect the matrix  $G_{com}$ .

Limitations (11) and (15) concern the location of widgets and applications on only one node. Assuming the matrix  $G_{com}$  is Boolean, these limitations are quickly checked using equality proof:

$$\sum_{i=1}^{Z+V} \left( \chi_1 \left( \sum_{j=1}^N G_{com}(i, j) \right) \right) = Z + V. \quad (21)$$

There is used the function of finding the presence of exactly a single one in the row or column of a Boolean matrix

$$\chi_1(\tau) = \begin{cases} 1, & \text{if } \tau = 1; \\ 0, & \text{if } \tau \neq 1. \end{cases} \quad (22)$$

In addition, the met of condition (21) corresponds to the fulfillment of the limitations on the mandatory allocation of all widgets and applications on the nodes of the computer system, which are given in limitations (13) and (14).

According to constraint (12), in mathematical modeling, only those nodes of the computer system that are involved in processing requests of the pool of widgets are considered. The check of this limitation can be quickly done by using the analysis of fragments of the columns of the resulting matrix by confirming the mandatory fulfillment of equality

$$\sum_{j=1}^N \left( \chi_2 \left( \sum_{i=1}^Z G_{com}(i, j) \right) \right) = N. \quad (23)$$

Here is a function of defining non-zero fragments of rows or columns of a Boolean matrix

$$\chi_2(\tau) = \begin{cases} 1, & \text{if } \tau \geq 1; \\ 0, & \text{else.} \end{cases} \quad (24)$$

This function can be used when checking the limitation (17) that all fragments of the database for serving requests for widgets should be distributed among the

nodes of the computer system, that is:

$$\sum_{j=1}^N \left( \chi_2 \left( \sum_{i=Z+V+1}^{Z+V+D} G_{com}(i, j) \right) \right) = N. \quad (25)$$

So, together with the developed mathematical model, it is proposed to apply in parallel an algorithm that determines whether the option under consideration complies with the given limitation:

1) vertical union of Boolean matrices G, H, S and obtaining a generalizing matrix  $G_{com}$ ;

2) verification of limitations for widgets and applications using expression (23);

3) verification of limitations for the nodes of a computer system using expression (24);

4) verification of limitations for the location of database fragments using expression (25).

### 3. Case study

The results obtained were tested in the synthesis of the control component of the UAV group, which is designed to conduct a survey of the area after an emergency. The user interface of the synthesized component includes a pool of widgets, which are divided into two groups:

1) widgets requiring prompt response to generated requests (group A);

2) widgets that allow a delay in the response to the generated request (group B).

The structure of the user interface is formed dynamically depending on the number of control objects. In particular, the pool of widgets can be divided into static and dynamic components. In the conducted experiments, the static component contained  $\aleph_{St} = 12$  widgets of group A. In the dynamic component, for each controlled object (UAV), the pool of widgets was increased by  $\aleph_{one} = 3$  widgets, which provided the formation of operational commands to control a specific widget. Therefore, the size of the pool of user interface widgets is determined by the following ratio:

$$\aleph_{pool} = \aleph_{St} + \aleph \cdot \aleph_{one}, \quad (26)$$

where  $\aleph$  – number of managed objects.

The standard response time to a request initiated by the operational widget of the pool that provides the user interface of this system for the synthesized component was determined as  $T_{spec} = 7$  seconds.

In synthesizing a component of a computer system, all distributed software components were distributed among the nodes of the software system. In particular, software was distributed that initiates the launch of the corresponding requests during activating a widget, applications, and database fragments to serve widget requests.

Two variants of the synthesis of the component were considered:

- using a standard model for the formation of information interconnections in a computer system;
- using the proposed model for the formation of information relationships in a computer system that is responsible for the control of moving objects.

The dependence of the average response time to a request on the number of UAVs in the considered group for the options under consideration is shown in Fig. 3.

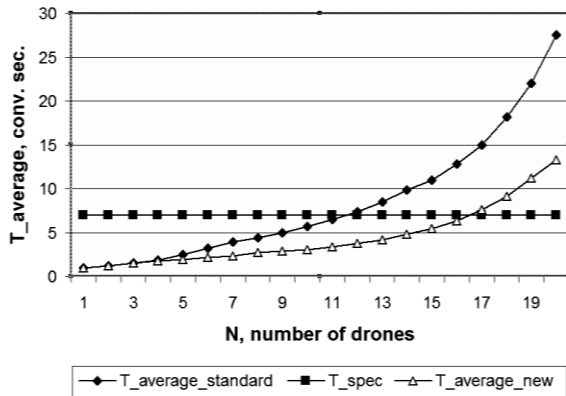


Fig. 3. Comparison of experimental results

As can be seen from the graphs in Fig. 3, when synthesizing the control component of a UAV constellation, which is designed to conduct a survey of the area after an emergency, the use of the standard model makes it possible to effectively support a constellation of up to 11 UAVs. The use of the proposed model made it possible to expand the grouping to 17 UAVs.

However, when the basic network is distorted, there is a need to carry out an operative redistribution of software across the nodes of the computer network, which initiates the launch of the corresponding requests when the widget, applications and fragments of the widget query service database are activated. In this case, the time of the transformation should also be referred to the most important parameters of this process, especially in cases of large dimensionality of the model (6) - (19). Therefore, the article proposed an algorithm for fast search for the space of feasible solutions. The results of redistribution during modeling for various variants of the formation of the control component of the UAV constellation with the use of the proposed algorithm and without it are shown in Fig. 4.

Fig. 4 shows that the fast algorithm for reallocation provides significant advantages over the standard algorithm in terms of execution time.

#### 4. Application perspectives

The proposed mathematical model of information interconnections in a computer system, which is respon-

sible for the control of moving objects, differs from the existing ones with such advantages:

1) the model is focused on the features of the functioning of the pool of widgets that support the user interface;

2) on the basis of the model, the fast algorithm for determining the space of admissible distributions on the nodes of the computer system of the software of widgets and applications, and distributed fragments of the database for servicing requests for widgets, has been developed.

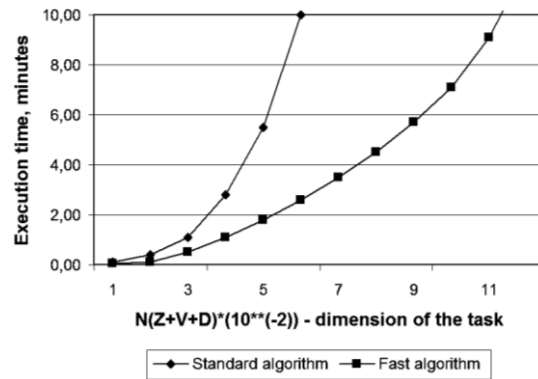


Fig. 4. Redistribution time

Due to these advantages, this model can be used when forming resource allocation queues between widget requests. But the most significant gain can be obtained when it is used to quickly redistribute the location of software between the nodes of a computer system in the event of local hardware failures. For this, it is necessary to develop a method for finding the optimal or rational redistribution of the location of software, the main criterion of which will be the solution of the optimization problem.

#### Conclusions

In this paper, a mathematical model was proposed that determines the parameters of information flows between the components of a computer system that are involved in the control of moving objects. The model uses the scheme of informational interconnections of the corresponding components in the environment of a computer system. The model is represented as a tuple of vectors, sets, Boolean matrices and a number of limitations. The limitations made it possible to take into account the specific features of the functioning of the pool of widgets that support the user interface when controlling movable objects. This model will allow for operational reallocation of resources by systems between requests using the proposed fast algorithm.

**The direction of further research** is the development of a method for finding the optimal or rational redistribution of the location of software, the main criterion of which will be the time to solve the optimization problem.

## References (GOST 7.1:2006)

1. Гордєєв, О. О. Моделі та оцінювання якості зручності використання інтерфейсу програмного забезпечення для людино-комп'ютерної взаємодії [Текст] / О. О. Гордєєв // *Радіоелектронні і комп'ютерні системи*. – 2020. – № 3 (95). – С. 84-96. DOI: 10.32620/reks.2020.3.09.

2. Kianpisheh, S. Cost-efficient server provisioning for deadline-constrained VNFs Chains: A parallel VNF processing approach [Text] / S. Kianpisheh, R. H. Glitho // *Proceeding of 2019 16th IEEE Annual Consumer Communications & Networking Conference*. – 2019. DOI: 10.1109/CCNC.2019.8651799.

3. Franti, P. Efficiency of random swap clustering [Text] / P. Franti // *Journal of Big Data*. – 2018. – Vol. 5, No. 13. – P. 1-29. DOI: 10.1186/s40537-018-0122-y.

4. Коваленко, Г. А. Методи синтезу інформаційної та технічної структури системи управління об'єктом критичного застосування [Текст] / Г. А. Коваленко, Г. А. Кучук // *Сучасні інформаційні системи*. – 2018. – Т. 2, № 1. – С. 22-27. DOI: 10.20998/2522-9052.2018.1.04.

5. Qiang, Ye. Distributed and adaptive medium access control for internet-of-things-enabled mobile networks [Text] / Ye. Qiang, W. Zhuang // *IEEE Internet of Things Journal*. – 2017. – Vol. 4, No. 2. – P. 446-460. DOI: 10.1109/JIOT.2016.2566659.

6. Kuchuk, G. Two-stage optimization of resource allocation for hybrid cloud data store [Text] / G. Kuchuk, S. Nechausov, V. Kharchenko // *International Conference on Information and Digital Technologies*. – Zilina, 2015. – P. 266-271. DOI: 10.1109/DT.2015.7222982.

7. Optimization of joint search and detection of objects in technical surveillance systems [Text] / H. Khudov, K. Tahyan, V. Cherpurnyi, I. Khizhnyak, K. Romanenko, A. Nevodnichii, O. Yakovenkon // *Advanced Information Systems*. – 2020. – Vol. 4, No. 2. – P. 156-162. DOI: 10.20998/2522-9052.2020.2.23.

8. Identification of the state of an object under conditions of fuzzy input data [Text] / S. Semenov, O. Sira, S. Gavrylenko, N. Kuchuk // *Eastern-European Journal of Enterprise Technologies*. – 2019. – Vol 1, No 4 (97). – P. 22-30. DOI: 10.15587/1729-4061.2019.157085.

9. Semenov, S. Testing process for penetration into computer systems mathematical model modification [Text] / S. Semenov, Cao Weilin // *Advanced Information Systems*. – 2020. – Vol. 4, No. 3. – P. 133-138. DOI: 10.20998/2522-9052.2020.3.19.

10. Фролов, В. В. Аналіз підходів до забезпечення безпеки хмарних сервісів [Текст] / В. В. Фролов // *Радіоелектронні і комп'ютерні системи*. – 2020. – № 1 (93). – С. 70-82. DOI: 10.32620/reks.2020.1.07.

11. Review and performance evaluation of FIFO, PQ, CQ, FQ, and WFQ algorithms in multimedia wireless sensor networks [Text] / H. Attar, M. R. Khosravi, S. I. Shmatkov, N. G. Kuchuk, M. Alhihi // *International*

*Journal of Distributed Sensor Networks*. – 2020. – No. 16(6). DOI: 10.1177/1550147720913233.

12. Franti, P. Efficiency of random swap clustering [Text] / P. Franti // *Journal of Big Data*. – 2018. – Vol. 5, No. 13. – P. 1-29. DOI: 10.1186/s40537-018-0122-y.

13. Improving big data centers energy efficiency: Traffic based model and method [Text] / G. Kuchuk, A. Kovalenko, I. E. Komari, A. Svyrydov, V. Kharchenko // *Studies in Systems, Decision and Control* / V. Kharchenko, Y. Kondratenko (Eds.). – Springer Nature Switzerland AG, 2019. – Vol. 171. – P. 161-183. DOI: 10.1007/978-3-030-00253-4\_8.

14. Nechausov, A. Synthesis of the air pollution level control system on the basis of hyperconvergent infrastructures [Text] / A. Nechausov, I. Mamusuć, N. Kuchuk // *Advanced Information Systems*. – 2017. – Vol. 1, No. 2. – P. 21-26. DOI: 10.20998/2522-9052.2017.2.04.

15. Resources Distribution Method of University e-learning on the Hypercovergent platform [Text] / V. Merlak, S. Smatkov, N. Kuchuk, A. Nechausov // *Conf. Proc. of 2018 IEEE 9th Int. Conf. on Dependable Systems, Service and Technologies. DESSERT'2018*. Kyiv, May 24-27. – 2018. – P. 136-140. DOI: 10.1109/DESSERT.2018.8409114.

16. Decomposition Method for Synthesizing the Computer System Architecture [Text] / V. Mukhin, N. Kuchuk, N. Kosenko, H. Kuchuk, V. Kosenko // *Advances in Intelligent Systems and Computing, AISC*. – 2020. – Vol. 938. – P. 289-300. DOI: 10.1007/978-3-030-16621-2\_27.

17. Тереник, Д. Порівняння SQL і NOSQL баз даних на прикладі проектування аффілейт репорт систем [Текст] / Д. Тереник, Г. А. Кучук // *Радіоелектронні і комп'ютерні системи*. – 2020. – № 1 (93). – С. 83-89. DOI: 10.32620/reks.2020.1.08.

18. Bulba, S. Composite application distribution methods [Text] / S. Bulba // *Advanced Information Systems*. – 2018. – Vol. 2, No. 3. – P. 128-131. DOI: 10.20998/2522-9052.2018.3.22.

## References (BSI)

1. Gordieiev, O. Modeli ta otsinyuvannya yakosti zruchnosti vykorystannya interfeysu prohramnoho zabezpechennya dlya lyudyno-kompyuternoyi vzayemodiyi [A models and assessment of quality of human-computer interaction software interface usability]. *Radioelektronni i komp'uterni sistemi – Radioelectronic and computer systems*, 2020, no. 3 (95), pp. 84-96. DOI: 10.32620/reks.2020.3.09.

2. Kianpisheh, S., Glitho, R. H. Cost-efficient server provisioning for deadline-constrained VNFs Chains: A parallel VNF processing approach. *Proceeding of 2019 16th IEEE Annual Consumer Communications & Networking Conference*, 2019. DOI: 10.1109/CCNC.2019.8651799.

3. Franti, P. Efficiency of random swap clustering. *Journal of Big Data*, 2018, vol. 5, no. 13, pp. 1-29. DOI: 10.1186/s40537-018-0122-y.

4. Kovalenko, A., Kuchuk H. Metody syntezy informatsiynoyi ta tekhnichnoyi struktur systemy upravlinnya obyektom krytychnoho zastosuvannya [Methods for synthesis of informational and technical structures of critical application object's control system]. *Advanced Information Systems*, 2018, vol. 2, no. 1, pp. 22-27. DOI: 10.20998/2522-9052.2018.1.04.
5. Qiang, Ye., Zhuang, W. Distributed and adaptive medium access control for internet-of-things-enabled mobile networks. *IEEE Internet of Things Journal*, 2017, vol. 4, no. 2, pp. 446-460. DOI: 10.1109/JIOT.2016.2566659.
6. Kuchuk, G., Nechausov, S., Kharchenko, V. Two-stage optimization of resource allocation for hybrid cloud data store. *International Conference on Information and Digital Technologies*, Zilina, 2015, pp. 266-271. DOI: 10.1109/DT.2015.7222982.
7. Khudov, H., Tahyan, K., Chepurnyi, V., Khizhnyak, I., Romanenko, K., Nevodnichii, A., Yakovenko, O. Optimization of joint search and detection of objects in technical surveillance systems. *Advanced Information Systems*, 2020, vol. 4, no. 2, pp. 156-162, doi: 10.20998/2522-9052.2020.2.23.
8. Semenov, S., Sira, O., Gavrylenko, S., Kuchuk, N. Identification of the state of an object under conditions of fuzzy input data. *Eastern-European Journal of Enterprise Technologies*, 2019, vol 1, no. 4 (97), pp. 22-30. DOI: 10.15587/1729-4061.2019.157085.
9. Semenov, S., Cao, Weilin. Testing process for penetration into computer systems mathematical model modification. *Advanced Information Systems*, 2020, vol. 4, no. 3, pp. 133-138. DOI: 10.20998/2522-9052.2020.3.19.
10. Frolov V. Analiz pidkhodiv do zabezpechennya bezpeky khmarnykh servisiv [Analysis of approaches providing security of cloud services]. *Radioelektronni i komp'uterni sistemi – Radioelectronic and computer systems*, 2020, no. 1 (93), pp. 70-82. DOI: 10.32620/reks.2020.1.07.
11. Attar, H., Khosravi, M. R., Shmatkov, S. I., Kuchuk, N. G., Alhihi, M. Review and performance evaluation of FIFO, PQ, CQ, FQ, and WFQ algorithms in multimedia wireless sensor networks. *International Journal of Distributed Sensor Networks*, 2020, no. 16(6). DOI: 10.1177/1550147720913233.
12. Franti, P. Efficiency of random swap clustering. *Journal of Big Data*, 2018, vol. 5, no. 13, pp. 1-29. DOI: 10.1186/s40537-018-0122-y.
13. Kuchuk, G., Kovalenko, A., Komari, I. E., Svyrydov, A., Kharchenko, V. Improving big data centers energy efficiency: Traffic based model and method. *Studies in Systems, Decision and Control*, 2019, vol. 171. Kharchenko, V., Kondratenko, Y. (Eds.). Springer Nature Switzerland AG, 2019, pp. 161-183. DOI: 10.1007/978-3-030-00253-4\_8.
14. Nechausov, A., Mamusuć, I., Kuchuk, N. Synthesis of the air pollution level control system on the basis of hyperconvergent infrastructures. *Advanced Information Systems*, 2017, vol. 1, no. 2, pp. 21-26. DOI: 10.20998/2522-9052.2017.2.04.
15. Merlak, V., Smatkov, S., Kuchuk, N., Nechausov, A. Resources Distribution Method of University e-learning on the Hyperconvergent platform. *Conf. Proc. of 2018 IEEE 9th Int. Conf. on Dependable Systems, Service and Technologies, DESSERT'2018*, Kyiv, 2018, pp. 136–140. DOI: 10.1109/DESSERT.2018.8409114.
16. Mukhin, V., Kuchuk, N., Kosenko, N., Kuchuk, H., Kosenko, V. Decomposition Method for Synthesizing the Computer System Architecture. *Advances in Intelligent Systems and Computing*, AISC, 2020, vol. 938, pp 289-300, DOI: 10.1007/978-3-030-16621-2\_27.
17. Terenyk, D., Kuchuk, H. Porivnyannya SQL i NOSQL baz danykh na prykladi proektuvannya affileyt report system [SQL & NOSQL database comparison by case designing affiliate system report]. *Radioelektronni i komp'uterni sistemi – Radioelectronic and computer systems*, 2020, no. 1 (93), pp. 83-89. DOI: 10.32620/reks.2020.1.08.
18. Bulba, S. Composite application distribution methods. *Advanced Information Systems*, 2018, vol. 2, no. 3, pp. 128-131, DOI: 10.20998/2522-9052.2018.3.22.

Надійшла до редакції 10.01.2021, розглянута на редколегії 16.02.2021

## МОДЕЛЮВАННЯ ІНФОРМАЦІЙНИХ ВЗАЄМОЗВ'ЯЗКІВ У КОМП'ЮТЕРНІЙ СИСТЕМІ УПРАВЛІННЯ РУХОМИМИ ОБ'ЄКТАМИ

Н. Г. Кучук, В. Ю. Мерлак

**Предмет дослідження** – пул віджетів, що забезпечує інтерфейс користувача у комп'ютерній системі управління рухомими об'єктами. **Об'єкт дослідження** – процес обміну інформацією між складовими комп'ютерної системи, що безпосередньо задіяні при управління рухомими об'єктами. Зазначено, що завдання розподілу ресурсів для пулу віджетів, які забезпечують оперативне управління рухомими об'єктами, є дуже важливою. При збільшенні кількості елементів пулу віджетів конкуренція за ресурси комп'ютерної системи суттєво зростає, відповідно зростають черги до найбільш затребуваних ресурсів. Тому необхідно провести перерозподіл ресурсів між запитами таким чином, щоб були враховані пріоритети запитів, а час реакції оперативних запитів не перевищував нормативний час. Такий розподіл неможливо реалізувати без моделі, яка б показала параметри всіх інформаційних потоків між задіяними складовими, що і є **метою даної статті**. При цьому модель повинна враховувати характерні особливості пулу віджетів, що підтримують інтерфейс користувача. **Результати дослідження**. Для досягнення мети вирішено декілька часткових завдань. Спочатку визначені основні множини, які будуть задіяні при моделюванні інформаційних взаємозв'язків між програмними компонентами, що обслуговують запити віджетів інтерфейсу користувача у середовищі комп'ютерної системи. Потім розроблена схема інформаційних взаємозв'язків між цими мно-



жинами. Надалі визначені та формалізовані параметри запитів, ініційованих віджетами, задіяних застосунків та фрагментів бази даних обслуговування запитів віджетів. Все це надало змогу сформувати модель інформаційних взаємозв'язків за допомогою кортежу отриманих векторів, множин та булевих матриць. У математичній моделі, що пропонується, накладає низка обмежень на елементи даного кортежу, пов'язані із характерними особливостями модуля управління інтерфейсом користувача комп'ютерної системи. Крім того, враховані обмеження, що пов'язані із функціонуванням базової мережі комп'ютерної системи. у кінцевій моделі булеві матриці, що впливають на розподіл ресурсів, визначені як змінні моделі. Запропоновано швидкий алгоритм визначення простору допустимих розподілів. **Висновок.** Розроблена математична модель визначає параметри інформаційних потоків між складовими комп'ютерної системи, що задіяні при управлінні рухомими об'єктами. Дана модель дозволить проводити оперативний перерозподіл ресурсів між запитами з використанням запропонованого швидкого алгоритму.

**Ключові слова:** пул віджетів; інтерфейс користувача; рухомий об'єкт; комп'ютерна система; інформаційні взаємозв'язки.

## МОДЕЛИРОВАНИЕ ИНФОРМАЦИОННЫХ ВЗАИМОСВЯЗЕЙ В КОМПЬЮТЕРНОЙ СИСТЕМЕ УПРАВЛЕНИЯ ПОДВИЖНЫМИ ОБЪЕКТАМИ

*Н. Г. Кучук, В. Ю. Мерлак*

Предмет исследования - пул виджетов, который обеспечивает интерфейс пользователя в компьютерной системе управления подвижными объектами. Объект исследования - процесс обмена информацией между составляющими компьютерной системы, которые непосредственно задействованы при управления подвижными объектами. Отмечено, что задача распределения ресурсов для пула виджетов, которые обеспечивают оперативное управление подвижными объектами, очень важна. При увеличении количества элементов пула виджетов конкуренция за ресурсы компьютерной системы существенно возрастает, соответственно растут очереди к наиболее востребованным ресурсам. Поэтому необходимо провести перераспределение ресурсов между запросами таким образом, чтобы были учтены приоритеты запросов, а время реакции оперативных запросов не превышало нормативное время. Такое распределение невозможно реализовать без модели, которая бы показала параметры всех информационных потоков между задействованными составляющими, что и является **целью данной статьи**. При этом модель должна учитывать характерные особенности пула виджетов, поддерживающие интерфейс пользователя. **Результаты исследования.** Для достижения цели было решено несколько частных задач. Сначала были определены основные множества, которые будут задействованы при моделировании информационных взаимосвязей между программными компонентами, которые обслуживают запросы виджетов интерфейса пользователя в среде компьютерной системы. Затем была разработана схема информационных взаимосвязей между этими множествами. Далее были определены и формализованы параметры запросов, инициированные виджетами, задействованных приложений и фрагментов базы данных обслуживания запросов виджетов. Все это дало возможность сформировать модель информационных взаимосвязей с помощью кортежа полученных векторов, множеств и булевых матриц. В предлагаемой математической модели накладывается ряд ограничений на элементы данного кортежа, связанные с характерными особенностями модуля управления интерфейсом пользователя компьютерной системы. Кроме того, учтены ограничения, связанные с функционированием базовой сети компьютерной системы. В конечной модели булевы матрицы, влияющие на распределение ресурсов, определенные как переменные модели. Предложено быстрый алгоритм определения пространства допустимых распределений. **Выводы.** Разработанная математическая модель определяет параметры информационных потоков между составляющими компьютерной системы, которые задействованы при управлении подвижными объектами. Данная модель позволит проводить оперативное перераспределение ресурсов между запросами с использованием предложенного быстрого алгоритма.

**Ключевые слова:** пул виджетов; интерфейс пользователя; движущийся объект; компьютерная система; информационные взаимосвязи.

**Кучук Ніна Георгіївна** – д-р техн. наук, доц., доц. каф. обчислювальної техніки та програмування, Національний технічний університет «Харківський політехнічний інститут», Харків, Україна.

**Мерлак Вікторія Юріївна** – асп. каф. комп'ютерних систем, мереж та кібербезпеки, Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут», Харків, Україна.

**Nina Kuchuk** – Doctor of Technical Sciences, Associate Professor, Associate Professor of Department of Computer Science and Programming, National Technical University “Kharkiv Polytechnic Institute”, Kharkiv, Ukraine;

e-mail: nina\_kuchuk@ukr.net, ORCID: 0000-0002-0784-1465.

**Viktoriia Merlak** – PhD student of Department of Computer Systems, Networks and Cybersecurity, National Aerospace University “Kharkiv Aviation Institute”, Kharkiv, Ukraine,

e-mail: v.merlak@csn.khai.edu, ORCID: 0000-0003-4096-0797.